

## Fall 2008 Handout 1

<i>Lecture</i>	<i>Monday</i>
1	September 28
2	October 5
3	October 12
4	October 19
5	October 26
6	November 2
7	November 9
8	November 16
9	November 23
10	November 30

### Reference book

*UNIX Power Tools, Second Edition* by Jerry Peek, Tim O'Reilly, and Mike Loukides; O'Reilly-Hall, 1997; ISBN 1-56592-260-3. It's \$59.95 in aisle 9 downstairs at the NYU Main Bookstore (not the NYU Computer Bookstore), 18 Washington Place, (212) 998-4667, <http://www.bookc.nyu.edu/main>. Page numbers in the handouts refer to this book.

The URL of the book is

<http://www.oreilly.com/catalog/upt2/>

### Computer bookstores in Manhattan

Computer Bookworks (212) 385-1616  
78 Reade Street (near City Hall), between Broadway and Church Street  
[bookman3@mindspring.com](mailto:bookman3@mindspring.com)

Barnes and Noble (212) 807-0099  
<http://www.barnesandnoble.com/>

### Contact information

Home page for this course: <http://i5.nyu.edu/~mm64/x52.9546/>

Mark Meretzky's email address: [mark.meretzky@nyu.edu](mailto:mark.meretzky@nyu.edu)

Mark Meretzky's home page: <http://i5.nyu.edu/~mm64/>

The system administrator's address is [comment@i5.nyu.edu](mailto:comment@i5.nyu.edu). For the NYU computer help desk, send email to [its.clientservices@nyu.edu](mailto:its.clientservices@nyu.edu), or call (212) 998-3333, or visit the ITS Client Services Center at <http://www.nyu.edu/its/askits/>. For problems with computer accounts and passwords, send email to the accounts office at [its.clientservices@nyu.edu](mailto:its.clientservices@nyu.edu) or call (212) 998-3333, or visit <http://www.nyu.edu/its/accounts/>. For information about grades, incompletes, and NYU courses, including courses which I will teach next semester, call the School of Continuing and Professional Studies at (212) 998-7190. To contact me after the course is over, please send me email—don't phone.

**Homework, exams, grades**

Your grade will depend on the homework you hand in to me on paper between the start of the course and 6:00 p.m. on the last class of the course. There will be no midterm or final. You get only one chance to hand in each problem. Hand in only the ones I assign in class, not all the problems in the Handouts.

Each assignment will be due one class after it is assigned. I can't predict when I will assign each assignment, since it depends on how fast the class goes. The class web page will list when each assignment is due. If you have handed in little or no work during the semester, it will not help your grade if you hand in all or most of it late at the end of the semester.

I will give you the answer to every assignment (except the individualized ones) on the day it is due. You therefore get no credit for homework that I receive after that date. If you will be absent on the due date, hand it in early or mail it to me so that I receive it early:

Mark Meretzky  
care of Joanne Davis  
NYU School of Continuing and Professional Studies  
Technology Division Administration  
10 Astor Place, room 505D (between Broadway and Lafayette Street; closed after 6:00 p.m.)  
New York, NY 10003-6935

Put your real name, course number, and section number on the homework. Do not email me your homework. If you miss a class, send someone to tape it or take notes, and to drop and pick up your homework.

I will return each assignment to you one class after you give it to me, except for homework which you give me on or after the last class. I will give you back that homework only if you give me a self-addressed stamped envelope.

I will not give a grade to each individual homework, but I will correct every mistake you make. If there are no corrections, you did the homework perfectly. The only grade you will receive will be the one you get for the entire course. There is no way to predict this grade before the end of the course, since it partially depends on how well everyone else in the class does. You may also be penalized for gross absenteeism.

**Collaboration**

To collaborate with one or two other people, you may collectively hand in one copy of every assignment with the names of the two or three authors. You must stay with the same partner(s) throughout the semester, and you will all receive the same grade. In the real world you will program with other people, so I encourage you to do so now.

You must do all your own work with no help from anyone except your partner(s), if any. ★ You will fail the course if I receive two copies of the same work from people who are not partners, or work on which you were helped by a person who is not your partner. After you're caught, it is too late to make the other person your partner. You will fail the course if you hand in copies of my answers, or anybody else's answers.

**The end of the course**

I will not tell you your grade. I always mail the grades to NYU immediately after the last class of the course, or the day I receive the grading sheet from NYU, whichever comes last. I don't know what NYU will do with the grades or how long they will take to make them available to you.

Carl F. Lebowitz Associate Dean of SCPS, says "Incomplete grades should be given only in rare circumstances where a student has been able to complete *nearly* all of the course assignments by the end of the semester." Some students request an Incomplete just to extend their computer account.

To extend your i5.nyu.edu account if you have requested an Incomplete, fill out form ITS 775 at

<http://www.nyu.edu/its/accounts/forms/request.extension.pdf>

and bring it to me by the last day of class. After I sign it, take it to the address at the top of the form. Do

not leave form ITS 775 with me.

To complete your incomplete, mail the “To complete an Incomplete” project in the class web page to me at the above address. Also include xeroxes of all the homework I returned to you during the semester, showing my comments. Include your full name, social security number or NYU ID number, email address, the course and section number, and the year and semester when you took the course. Do not email me your late homework.

### Computer labs at NYU:

<http://www.nyu.edu/its/labs/>

(212) 998-3409	Room LC-8 Tisch Hall, two flights down
PC's	40 West 4th Street at Greene Street
printer:	none accessible via the i5.nyu.edu <b>lpr</b> command

(212) 998-3457	14 Washington Place, one flight down
PC's	between Greene and Mercer Streets
printer:	none accessible via the i5.nyu.edu <b>lpr</b> command

(212) 998-3421	Education Building, second floor
Macs	35 West 4th Street at Greene Street
printer:	<b>edlab</b>

(212) 998-3504	North Dorm, two flights down
PC's and Macs	75 Third Avenue (southeast corner of Third Avenue & 12th Street)
printer:	<b>ndlab</b>

To get your plastic, magnetic NYU ID card, see

<http://www.nyu.edu/nyucard/>

### Your i5.nyu.edu account

Our computer is Sun 250 server running Solaris 10 (SunOS 5.10). It is actually a Solaris *zone*, a virtual machine sitting on top of a “global” zone. Its Internet hostname is **i5.nyu.edu** and its IP version 4 address is **128.122.253.152**.

Your i5.nyu.edu login name is listed at

<http://i5.nyu.edu/~mm64/common/students.html>

It is the same as your NYU NetID used by your NYU DIAL or NYU Home account. It will be two or three *lowercase* letters (your initials) followed by one or more digits. In these Handouts, we'll assume your login name is **abc1234**.

Your i5.nyu.edu secret password is the same as your central NYU single sign-on password used by your NYU DIAL or NYU Home account. In these Handouts, we'll assume your password is **Bacall18?**.

Before using your login name and password for the first time, register at <http://start.nyu.edu/>. First time i5.nyu.edu users must leave the password field blank as they have not yet set their password. They will then be prompted to enter their social security number and birth date.

To change your password to a more colorful one, e.g., **bogart!** or **Bacall18?**, go back to <http://start.nyu.edu/>.

### The “secure shell” ssh

To log into **i5.nyu.edu**, you have to run a program that speaks the “secure shell protocol”. One example is a program named **PuTTY** or **putty.exe**. If you don't already have it, get it from

`http://www.chiark.greenend.org.uk/~sgtatham/putty/`  
`https://www.nyu.edu/its/software/`

(1) On Windows, run `putty.exe`. A window named **PuTTY Configuration** will appear. Type `i5.nyu.edu` as the host name, and select the radio button for the protocol **SSH**. The port number should be 22. Then press **Open**. Dismiss any **PuTTY Security Alert** window that may appear.

`login as: abc1234`  
`Password: Bacall18?`

To copy from the **PuTTY** window, select (highlight) the text you want to copy. You do not have to say “copy”. To paste into the window, click on the place where you want to paste and press the right mouse button.

(2) From any other Unix host, you can get to `i5.nyu.edu` by running the program `ssh`. If you don’t already have it, get it from

`http://www.openssh.com/`

On Mac OSX, for example, launch the **Terminal** application to get a Unix shell window. Pull down the **Font** menu and select a pleasant font. Then give the command

`ssh abc1234@i5.nyu.edu`

### After logging in

When you are finished logging in, you will see the prompt. To verify that you are really logged in, run simple programs such as

`1$ date` *Press RETURN after each command line.*  
`2$ cal 1 2008` *Need space before each command line argument.*

### Log out

`1$ exit` *or logout on other systems*

If the terminal window is still open, pull down the **File** menu and select **Exit** or **Quit** to close it.

### The special keys: symptoms and antidotes

(1) If you accidentally type `control-z`, it will say **Stopped**. To start things up again, type

`1$ fg` *Bring the most recently stopped program back into the foreground.*

If it says **You have stopped jobs** when you try to log out, type `fg` to give your stopped job a chance to finish. Repeat if necessary.

(2) Press **backspace**, **delete**, or `←` to erase the last character typed. If your `←` or **backspace** key doesn’t work, type

`2$ stty erase ←`

(with a space before the `←`) and press **RETURN**. As a last resort, see if `control-h` will backspace.

(3) To kill a long program, type `control-c` on a terminal or a PC. You may have to type it several times.

(4) `Control-s` will freeze the screen; unfreeze it with `control-q`. Similarly, **Hold Screen** on a terminal or **Scroll Lock** on a PC will freeze the screen. Unfreeze it with another **Hold Screen** or **Scroll Lock**.

(5) Never press **Caps Lock** in Unix: almost everything we type will be lowercase. Don’t confuse

(5a) the lowercase letter **l**, the uppercase letter **I**, and the digit **1**

(5b) the lowercase letter **o**, the uppercase letter **O**, and the digit **0**

- (5c) the diagonal slash / and the backslash \
- (5d) the single quote ', the double quote ", and the back quote `
- (5e) the dash -, the underscore \_, and the tilde ~
- (5f) the left parentheses (, the left curly brace {, and the left square bracket [
- (5g) the right parentheses ), the right curly brace }, and the right square bracket ]
- (5h) the vertical bar (pipe symbol) |, the colon :, and the exclamation point !
- (5i) any uppercase letter and the corresponding lowercase letter.

### Syllabus for Unix Tools X52.9546

(1) Unix Operating System X52.9545 dealt with general-purpose tools for manipulating text: editing, sorting, searching, copying, comparing, merging, collating, formatting, and printing. Unix Tools X52.9546 will continue along these lines.

```
1$ whatis awk compress file find ftp ln sort tar tr
find          find (3c++/3)- Finds an occurrence of value in a sequence.
sort          sort (3c++/3)- A templated algorithm for sorting collections of entities.
```

(2) We'll cover tools to help you program in C and C++. You will therefore have to do a lot of C programming in this course.

```
3$ whatis cc CC nm strip size time dbx ar rcs make
cc            cc (1)- C compiler
CC            CC (1)- C++ compiler
time          time (3f)- return system time
dbx           dbx (1) - source-level debugging tool
```

(3) Some programs come with their own command languages, e.g. **emacs** and Macintosh Hypercard. **Tcl** is an *embeddable* or transplantable language that can be added to any program that has only a rudimentary command language, e.g., **mail**, **ftp**, **telnet**, etc.

**Tk** is an extended form of **Tcl** containing commands for writing GUI interfaces. **Expect** is another extended form of **Tcl** for writing interactive shells: it convinces other programs that they're dealing with a live human being at a terminal rather than with a shellscript.

### Three ways to discover which shell you're using

The Bourne and Korn shells execute the commands in a file named **.profile** when you login in, but the C shell executes the commands in a file named **.login** when you login in (Handout 2, p. 2). That's why you must know which shell you're using. In either case, the file must be in your home directory.

**ps** is the Unix equivalent of right clicking on the bottom bar in Windows and selecting **Task Manager...** → **Processes** (or going to the **Command Prompt** and running **tasklist** to see the PID numbers).

```
1$ echo $SHELL
2$ grep abc1234 /etc/passwd
3$ ps | more
```

*Look at the seventh field, p. 53.*

*"Process status": list all the programs you're running, p. 34.*

### Create a Korn shell .profile file: pp. 35–38

Your **.profile** file should be a copy of the file  
`~mm64/public_html/x52.9545/src/.profile`:

```
1$ cd
2$ pwd
```

*Go to your home directory.*

*Make sure you arrived there.*

```

3$ ls -l | more           all names except those that start with a dot
4$ ls -la | more         all names, including those that start with a dot (Los Angeles)

5$ mv .profile old.profile   Rename your existing .profile file, if you have one.
6$ ls -la | more

```

Copy `~mm64/public_html/x52.9545/src/.profile` into your current directory, which is now your home directory. For the dot which is the second argument of the `cp`, see Handout 1, p. 9. The other dot is merely part of the filename `.profile`; it does not mean “the current directory”.

```

7$ cp ~mm64/public_html/x52.9545/src/.profile .
8$ ls -la | more

```

`cmp` will give you dead silence if the two files are identical, p. 20:

```

9$ cmp ~mm64/public_html/x52.9545/src/.profile .profile
10$ exit           The commands in your .profile will be executed when you log back in.

```

If you change the contents of your `.profile` file, you must log out and log back in. The commands in your `.profile` are executed only when you log in, not when you put them into the file.

### The contents of your Korn shell `.profile` file

No backslash is needed to split a long command immediately after a pipe (lines 81–83). See pp. 107–108.

—On the Web at

<http://i5.nyu.edu/~mm64/x52.9545/src/.profile>

```

1 #This file is named .profile
2 #The Korn shell executes the commands in this file when you log in.
3 #Page numbers in this file refer to the ksh(1) manual page.
4
5 #"export" (p. 41) creates an environment variable (pp. 23-24).
6
7 #Directories shell searches for executables: pp. 16, 27. /usr/ucb for whereis
8 export PATH=$PATH:/usr/ucb:.
9
10 #The -t option of man makes man run this program.
11 #~mm64 is the full pathname of mm64's home directory: pp. 4-5.
12 export TCAT=~mm64/bin/tcat
13
14 #The editor to be invoked by mail, mailx, dbx, etc: p. 14.
15 export EDITOR=/bin/vi
16
17 #Make vi display the words INSERT MODE while you're in insert mode.
18 export EXINIT='set showmode' #or export EXINIT='set showmode number'
19
20 #The name of the computer.
21 export HOST='hostname'
22
23 #The Korn shell will check this file for new mail every 10 minutes: p. 16.
24 export MAIL=/var/mail/$LOGNAME
25
26 #Default printer for the lpr, lpq, lprm programs.
27 export PRINTER=edlab
28
29 #Source code directory is on the web at http://i5.nyu.edu/~mm64/x52.9545/src/

```

```
30 export S45=~mm64/public_html/x52.9545/src
31
32 #if necessary to make your screen editor (vi, emacs, pico) work properly
33 export TERM=vt100
34
35 #Make the command numbers in the prompt start at 1 each time you log in:
36 #p. 27 for .sh_history. ~ is the full pathname of your home directory: pp. 4-5.
37 rm -f ~/.sh_history
38
39 #Prompt string one. See pp. 10-12 for chopping, p. 16 for !.
40 #The %%.* chops off the end of $HOST: i5.nyu.edu becomes i5
41 #The ##*/ chops off the beginning of $PWD: /home1/a/abc1234 becomes abc1234
42 PS1='! ${HOST%%.*}:${PWD##*/} $ '
43
44 #The following command must come *after* the export EDITOR=/bin/vi.
45 #otherwise the export EDITOR=/bin/vi would turn emacs mode back off (p. 12).
46
47 #Retrieve and edit previous commands with emacs, pp. 28-32.
48 #See pp. 46-50 for set.
49 set -o emacs
50
51 #Make the 4 arrow keys work in emacs mode, pp. 28-32. Only in ksh93, sorry.
52 /bin/loadkeys ~mm64/public_html/x52.9545/src/loadkeys_set
53
54 #Don't let control-d log you out (p. 48): must type "exit" (p. 41).
55 set -o ignoreeof
56
57 #Don't let > destroy the existing contents of a file: pp. 22, 49.
58 set -o noclobber
59
60 #For alias, see pp. 3-4, 39.
61 #"history" is itself an alias for "fc -l", pp. 4, 42.
62 alias h=history
63
64 #See all the jobs you're running, pp. 25, 43.
65 alias j=jobs
66
67 #/bin/mailx is better than /bin/mail. Used in line 81.
68 alias mail=mailx
69
70 #Turn off last 2 r and w bits of a newborn file. For example, date > newborn
71 #will create a newborn file whose bits are rw-----. The octal argument of
72 #the umask command has a 1 in each bit that should be turned off. See p. 55.
73 umask 077
74
75 #Let other people send you messages with the talk program.
76 #It turns on the two rightmost w bits of your terminal.
77 mesg y
78
79 #See new and unread mail in alphabetical order of who sent it.
80 #Must say mailx because an alias works only if it's the first word of a command.
81 mail -e && mailx -H |
82 grep '^.[NU]..[0-9]' |
83 sort +0.6f -0.25 +0.6 -0.25 +0.2n
```

```

84
85 #date
86 #cal
87 #who | grep def5678
88 #~mm64/bin/moon

```

### Create a Korn shell alias:

```

1$ alias g=grep           Create an alias named g. No space around the equal sign.
2$ alias g                Verify that you created it.
3$ alias | more          See all your alias's.

4$ g atlantic /usr/dict/words
atlantic
transatlantic

5$ unalias g              Remove the alias g.
6$ alias g                Verify that it's gone.

```

### An alias that needs single quotes

An alias needs single quotes when it contains more than one word, i.e., when it contains white space:

```

1$ alias g='grep -i'
2$ alias g
3$ g atlantic /usr/dict/words
Atlantic
atlantic
Atlantica
transatlantic

```

An alias also needs single quotes when it contains the name of more than one program, i.e., when it contains a semicolon or pipe:

```

4$ alias g='date; cal; who | grep abc1234'
5$ alias g
6$ g

```

### Aliases vs. variables

An alias is easier to use than a variable: you don't have to type the dollar sign.

```

1$ alias g=grep           Create an alias named g.
2$ g atlantic /usr/dict/words doesn't need a dollar sign

3$ g=grep                Create a variable named g.
4$ $g atlantic /usr/dict/words needs the dollar sign

```

But an alias can be used only as the *first* word of a command, while a variable can be used anywhere in a command:

```

5$ cd $S45                The variable $S45 will contain a directory name.

```

Use an alias for the name of a command; use a variable for the name of a file or directory.



### Configure the Apache Web Server

When anyone in the world points their web browser at a web page stored on our host `i5.nyu.edu`, the program on `i5.nyu.edu` that sends them a copy of the page is called the *web server*. The name of this program is `/usr/apache/bin/httpd` (“Hyperterxt Transport Protocol Dæmon”). Here are excerpts from its configuration file `/etc/apache/httpd.conf`. The documentation for the directives in this file is at

<http://www.apache.org/docs/mod/directives.html>

```

1 #If a URL specifies no filename, use this filename by default.
2 #If the directory has no file with this name, do an ls -l of the
3 #directory instead.
4
5 DirectoryIndex index.html index.php
6
7 #In a URL, a user's loginname with a ~ in front of it stands
8 #for the following subdirectory of that user's home directory.
9
10 UserDir public_html
11
12 #The following is automatically prefixed to the directory specified
13 #in a URL, unless the specified directory begins with a user's
14 #loginname with a ~ in front of it.
15
16 DocumentRoot "/var/apache/htdocs"
17
18 #In the URL of a gateway, the following short string stands
19 #for the much longer string.
20
21 ScriptAlias /cgi-bin/ "/var/apache/cgi-bin/"

```

#### ▼ Homework 1.1: create a World Wide Web home page

Create a file named `index.html` in the `public_html` subdirectory of your home directory. You can write your own, or copy the one in `/$S45/public_html` and modify it.

```

1$ cd ~/public_html
2$ pwd

3$ cp $S45/public_html/index.html .
4$ ls -l index.html

5$ chmod 644 index.html           turn on all three r's: rw-r--r--
6$ ls -l index.html

7$ vi index.html                  Change it to describe yourself.

```

Do not write the line numbers or the blank after each line number. `chmod` your `index.html` file to `rw-r--r--`. `chmod` your home directory and its `public_html` subdirectory to `rw-r-xr-x` if they do not already have these permissions (X52.9545 Homeworks 1.2 and 1.3).

Then tell your friends that your World Wide Web URL is `http://i5.nyu.edu/~abc1234/` where `abc1234` is your `i5.nyu.edu` login name, and put a note to this effect in your `.plan` file. In Unix, `~abc1234` stands for `abc1234`'s home directory. In a URL, `~abc1234` stands for the `public_html` subdirectory of `abc1234`'s home directory because of the `UserDir` in `httpd.conf`. The default file-name in a URL is `index.html` because of the `DirectoryIndex` in `httpd.conf`.

If you display your home page with a browser, and then re-edit it while the browser is still running, press the browser's **Reload** button to update the display.

To see which machines have accessed your home page,

```
8$ cd /var/apache/logs
9$ pwd
```

```
10$ ls -l access_log How many characters?
-rw-r--r-- 1 nobody bin 3227414 Jan 25 12:16 access_log
```

```
11$ wc -l access_log How many lines? Be patient.
33353 access_log
```

```
12$ grep abc1234 access_log | more Be patient.
```

For the web pages of the other students in the class, see

```
http://i5.nyu.edu/~mm64/common/students.html class roster
http://i5.nyu.edu/~mm64/x52.9545/00120083.html class photo
```

See the 10-minute guide to HTML in the HTML home page

```
http://www.w3.org/MarkUp/
```

—On the Web at

```
http://i5.nyu.edu/~mm64/x52.9545/src/public_html/index.html
```

```
1 <HTML>
2 <HEAD>
3 <TITLE>These words are displayed in window's title bar.</TITLE>
4 </HEAD>
5
6 <BODY>
7 <H1>These words are displayed in big letters in the window itself.</H1>
8
9 <H2>Pairs of tags</H2>
10 <P>
11 Surround each tag with "less than" and "greater than" signs.
12 Do not write these characters for any other purpose.
13 If you really must say "less than" and "greater than",
14 say &lt; and &gt; and don't forget the semicolons.
15 Write an ampersand as &amp;.
16 </P>
17
18 <P>
19 This is the start of a new paragraph.
20 The words within the tags are case-insensitive, except if they're in quotes.
21 Most tags come in pairs.
22 The second tag of the pair always starts with a slash.
23 For example,
24 surround the entire page with a pair of HTML tags (lines 1 and 148).
25 Every page consists of a head and a body,
26 each enclosed by its own pair of tags.
27 </P>
28
29 <P>
30 One tag that does not come in pairs
```

```
31 is the HR tag that draws a horizontal rule (line).
32 <HR>
33 <!-- A comment has an exclamation point, four dashes, and two spaces. -->
34 </P>
35
36 <H2>Headers</H2>
37 <P>
38 The H1 tags make a big header;
39 the H2 and H3 tags make smaller headers.
40 The TITLE tags enclose the title of the window that displays the page.
41 The TITLE should be shorter than the H1.
42 Put the TITLE tags in the head,
43 and the H1 tags in the body.
44 </P>
45
46 <H2>Special fonts</H2>
47 <P>
48 You can apply
49 <EM>emphasis</EM>,
50 <STRONG>strong emphasis</STRONG>,
51 or
52 <FONT SIZE = 7>size</FONT>.
53 </P>
54
55 <H2>Spacing</H2>
56 <P>
57 To make the file easier to edit,
58 put each phrase on a line by itself
59 and skip lines for legibility.
60 The browser will pack the text together
61 in civilized paragraphs when displaying it on the screen.
62 </P>
63
64 <P>
65 Use the "break" tag between lines of a poem:
66 </P>
67
68 <P>
69 Two households, both alike in dignity,
70 <BR>
71 In fair Verona, where we lay our scene,
72 <BR>
73 From ancient grudge break to new mutiny,
74 <BR>
75 Where civil blood makes civil hands unclean.
76 </P>
77
78 <P>
79 "Preformat" text to preserve its indentation and relative position:
80 </P>
81
82 <PRE>
83 From forth the fatal loins of these two foes
84   A pair of star-cross'd lovers take their life;
```

```

85 Whose misadventured piteous overthrows
86   Do with their death bury their parents' strife.
87 </PRE>
88
89 <H2>Lists</H2>
90 <UL>
91 <LI>Surround the entire list with a pair of UL tags.</LI>
92 <LI>Surround each item with a pair of LI tags.</LI>
93 <LI>To get numbers instead of bullets, change the UL's to OL's.</LI>
94 </UL>
95
96 <H2>Tables</H2>
97 <P>
98 Surround the entire table with a pair of TABLE tags (lines 105 and 123).
99 Within the table, surround each row with a pair of TR tags.
100 Within each row, surround each item of data with a pair of TD tags.
101 Before the first row and/or at the start of each row,
102 surround each table header with a pair of TH tags.
103 </P>
104
105 <TABLE BORDER>
106 <TR>
107 <TH></TH>
108 <TH>Column 1</TH>
109 <TH>Column 2</TH>
110 </TR>
111
112 <TR>
113 <TH>Row 1</TH>
114 <TD>row 1, col 1</TD>
115 <TD>row 1, col 2</TD>
116 </TR>
117
118 <TR>
119 <TH>Row 2</TH>
120 <TD>row 2, col 1</TD>
121 <TD>row 2, col 2</TD>
122 </TR>
123 </TABLE>
124
125 <H2>Special Characters</H2>
126 <OL>
127 <LI>&uuml;mlaut, &ccedil;edilla</LI>
128
129 <LI>Arabic: "Haza modhish!" ("It's amazing!",
130 from the Berlitz Phrasebook)
131 <SPAN LANG = "AR">&#x0647;&#x0632;&#x0627;
132 &#x0645;&#x062F;&#x0647;&#x0634;</SPAN></LI>
133
134 <LI>Hebrew: "Kosher"
135 <SPAN LANG = "HE">&#x05DB;&#x05BC;&#x05B8;&#x05E9;&#x05C1;&#x05B5;&#x05E8;</SPAN></LI>
136
137 <LI>Russian: "winter forest"
138 <SPAN LANG = "RU">&#x0437;&#x0438;&#x043C;&#x043D;&#x0438;&#x0439;

```

```

139 &#x043B;&#x0435;&#x0441;</SPAN></LI>
140
141 <LI>Chinese characters for "up", "down", "stream":
142 <SPAN LANG = "ZH">&#x4E0A; &#x4E0B; &#x5DDD;</SPAN></LI>
143 </OL>
144 </BODY>
145 </HTML>

```

	Column 1	Column 2
Row 1	row 1, col 1	row 1, col 2
Row 2	row 2, col 1	row 2, col 2

## Color

**BGCOLOR** is the background color of the page. **LINK** is the color of links that you have not yet visited; **VLINK** is the color of links that you have visited.

```
<BODY BGCOLOR = "000000" TEXT = "00ffff" LINK = "ffffff" VLINK = "ffff00">
```

It's easier to catch errors if you line them up like this:

```

<BODY
BGCOLOR = "000000"
  TEXT = "00ffff"
  LINK = "ffffff"
  VLINK = "ffff00">

```

Each pair of hexadecimal digits represents a number in the range 0 to 255 inclusive. The three numbers represent the amounts of red, green, and blue. For example, **black** is **000000**, white is **ffffff**, and yellow is a mixture of red and green: **ffff00**. To see examples of colors in decimal,

```

1$ grep -i yellow /usr/openwin/lib/rgb.txt | head -5  uppercase X eleven
173 255 47      green yellow
173 255 47      GreenYellow
154 205 50      yellow green
154 205 50      YellowGreen
250 250 210     light goldenrod yellow

```

```

#!/bin/sh
#Output the yellows as 24-bit values in hexadecimal.

awk '
  /[Yy]ellow/ {
    printf "%02x%02x%02x\t", $1, $2, $3
    for (i = 4; i <= NF; ++i) {
      if (i > 4) {
        printf " "
      }
      printf "%s", $i
    }
    printf "\n"
  }
' /usr/openwin/lib/rgb.txt

```

```
adff2f  green yellow
adff2f  GreenYellow
9acd32  yellow green
9acd32  YellowGreen
fafad2  light goldenrod yellow
```

```
#!/bin/sh
#Output the yellows as 24-bit values in hexadecimal.

perl -ane '
    if (/yellow/i) {
        printf "%02x%02x%02x", $F[0], $F[1], $F[2];
        print "\t@F[3 .. $#F]\n";
    }
' /usr/openwin/lib/rgb.txt
```

### Accent marks, special characters, other character sets

For the ümlaut, çedilla, etc. (line 127), don't forget the leading ampersand and trailing semicolon. All the special characters are at

<http://www.w3.org/TR/REC-html40/sgml/entities.html>

Look up the 16-bit hexadecimal code for each foreign character in the code charts at <http://www.unicode.org/>.



### Compile a C or C++ program

i5 has the following compilers:

```
cc      C
gcc     GNU C
g++    GNU C++
```

We'll use **gcc** as our example throughout. Give only the name of the **.c** file, not the names of the **.h** files, as a command line argument to **gcc**.

```
1$ gcc prog.c           Create an executable file named a.out
2$ ls -l a.out         a.out has its x bits turned on.
3$ a.out              Execute the C program.
4$ mv a.out prog      Rename the a.out file prog

5$ gcc -o prog prog.c  Create an executable file named prog
6$ ls -l prog

7$ gcc -o ~/bin/prog prog.c
8$ ls -l ~/bin
```

**\${1%.c}** is the shellscript's first command line argument, with the trailing **.c** chopped off.

```
#!/bin/ksh

/usr/local/bin/gcc -o ~/bin/${1%.c} $*
```

**<Angle brackets> in an #include directive**

The `/usr/include` directory and its descendants contain the `.h` files that will be `#include`'d by many programs: `stdio.h`, `stdlib.h`, `math.h`, etc. Do not specify the full path name of these `.h` files when you `#include` them. Enclose them in `<angle brackets>`, which will make the compiler automatically add `/usr/include` to the start of their names before searching for them. Never write a full pathname within angle brackets.

`CC include` files are in the directory `/usr/include/CC`.

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <sys/types.h>
```

```
1$ cd /usr/include
```

```
2$ ls -l | more
```

```
-rw-r--r--  1 root    bin           9606 Nov 23  2004 math.h
-rw-r--r--  1 root    bin          11853 Jan 21  2005 stdio.h
-rw-r--r--  1 root    bin           9296 Jan 21  2005 stdlib.h
```

```
3$ ls *.h | wc -l
    261
```

```
4$ cd sys
```

```
5$ pwd
```

```
/usr/include/sys
```

```
6$ ls -l | more
```

```
-rw-r--r--  1 root    bin          17028 Dec 21  2006 types.h
```

**"Double quotes" in an #include directive**

If you do not want the compiler to add `/usr/include` to the start of the `.h` file's name, use "double quotes" instead of `<angle brackets>`. For example, to `#include` a `.h` file that will be used by only a few C programs,

```
#include "moon.h"                /* relative pathname */
#include "/home1/m/mm64/46/moon/moon.h" /* full pathname */
```

If you're always going to be in the directory that contains the `.h` file when you give the `gcc` command, you can write the relative pathname. Otherwise, write the full pathname.

**The -I option of gcc**

Suppose the file `stdlib.h` was in an unusual place on your machine, e.g., the directory `/usr/exclude` instead of `/usr/include`. You could change all of your C programs from

```
#include <stdlib.h>
```

to

```
#include "/usr/exclude/stdlib.h"
```

but then you would have to change them back when you port the program to a new machine.

A better solution would be to keep the `#include <stdlib.h>`, but use the `-I` option of `gcc` to tell the C compiler to try to add another directory name in addition to `/usr/include` to the start of an angle-bracketed name.

```
1$ gcc -I/usr/exclude -o prog prog.c
```

*no space after the I*

You can write several `-I` options, each one naming another directory.

```
2$ gcc -I/usr/exclude -I.. -o prog prog.c
```

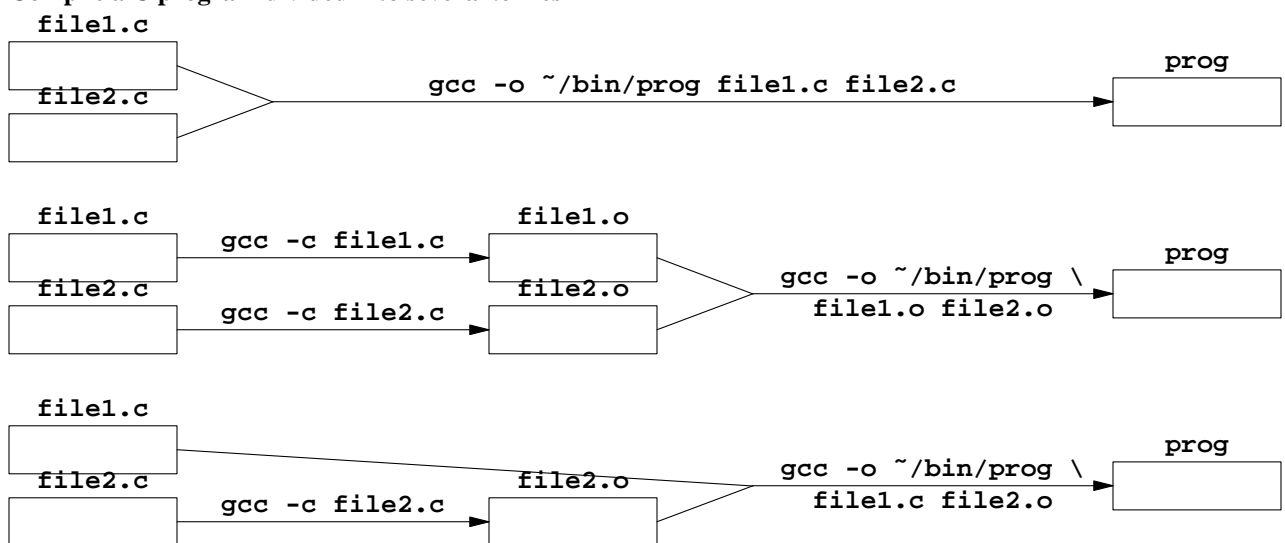
*.. is current dir's parent*

The `-I` option does not tell `gcc` where to find `.c` files; it tells `gcc` where to find only the angle-bracketed `.h` files.

### Intermediate steps in a C compilation

```
1$ gcc -E prog.c > prog.i           Create prog.i
2$ gcc -S prog.c                   Create prog.s
3$ gcc -c prog.c                   Create prog.o
```

### Compile a C program divided into several .c files



Give the `.c` files as command line arguments to `gcc` in any order. Do not give the names of the `.h` files as command line arguments to `gcc`.

```
1$ gcc -o ~/bin/prog file1.c file2.c
```

Compile the `.c` files individually and then link them together:

```
2$ gcc -c file1.c           Create file1.o
3$ gcc -c file2.c           Create file2.o
4$ gcc -o ~/bin/prog file1.o file2.o
```

You can even give `gcc` a mixture of `.c` and `.o` files:

```
5$ gcc -c file2.c           Create file2.o
6$ gcc -o ~/bin/prog file1.c file2.o
```

### Libraries

If your program calls math library functions such as `sqrt`, `sin`, `cos`, etc., add the `-lm` option (minus lowercase LM) to the end of the `gcc` command.

When you specify a library with this `-l` option, the name of the file that contains the library is whatever follows the `-l`, with a `lib` added to the front and a `.a` added to the end. For example, the name of file that contains the library you specified with the `-l` option is `libm.a`.



By default, **gcc** assumes that every library is in the **/usr/lib** directory; you can specify a different directory with a **-L** option before the **-l** option.

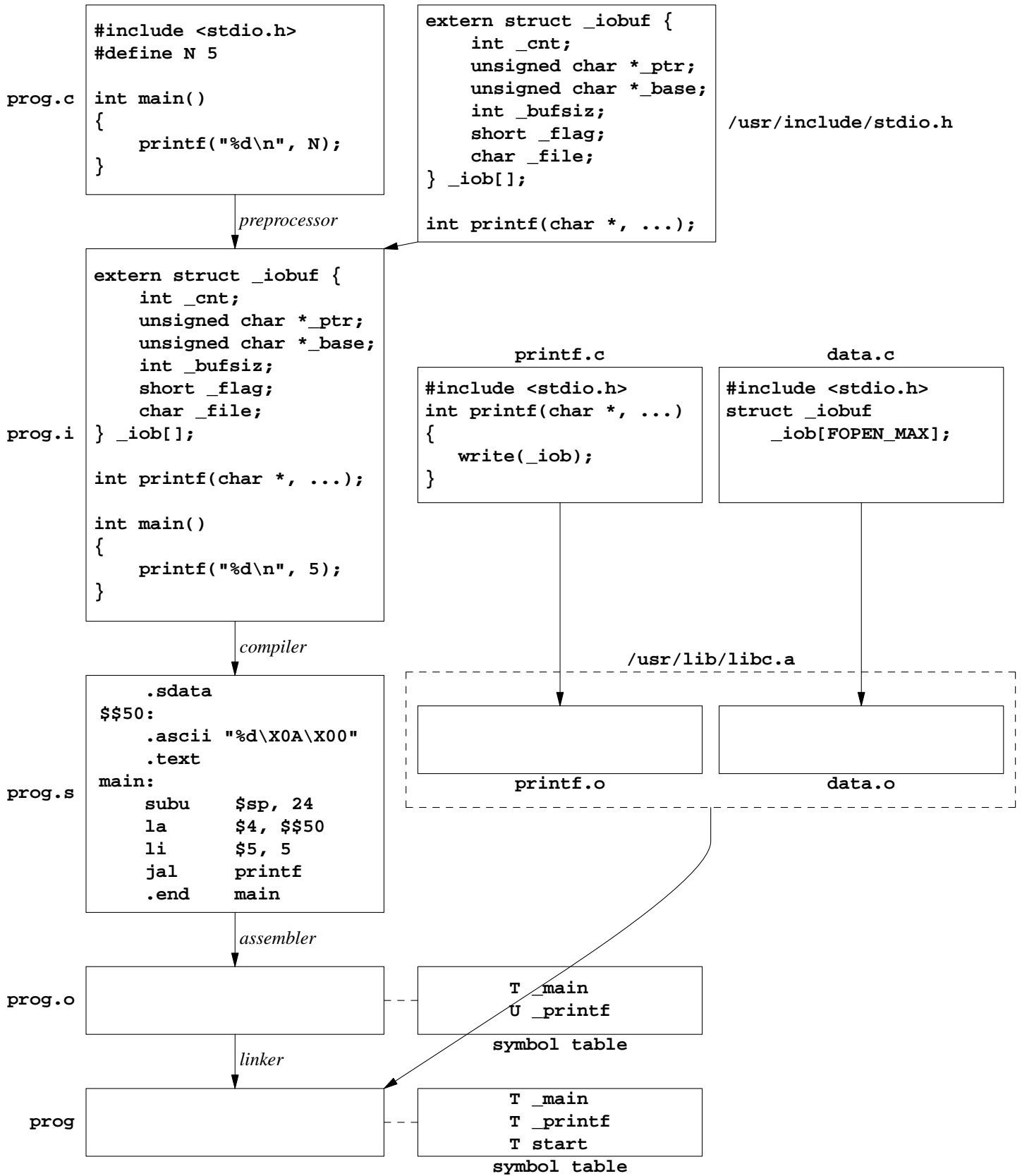
▼ **Homework 1.2: create an executable moon**

Copy the **.h** and **.c** files whose names start with **moon** from the directory **\$m46/moon** to your **bin** directory. Compile them into an executable file named **moon**. Put the **-lm** option (“math library”, minus lowercase LM) at the end of the **gcc** command line that links **moon** together.

```
1$ moon                                today
2$ moon 25 1 2008                       day, month, year
```



The file **/usr/include/stdio.h** contains the declarations for functions such as **printf** and **scanf**. See pp. 155–156 in K&R for declarations of functions that take a variable number of arguments.



### The School of Continuing Education is now the School of Continuing and Professional Studies

And its URL changed from `http://www.sce.nyu.edu/` to `http://www.scps.nyu.edu/`  
`grep -E` means `egrep`. `egrep` knows about the “or” symbol `|`, but plain old `grep` doesn’t.

If you give more than one input file to `grep` or `egrep`, each line of output will begin with the name of the input file where the line was found:

```
1$ grep top oldsmoky
On top of Old Smoky

2$ grep top oldsmokey spaghetti
oldsmokey:On top of Old Smoky
spaghetti:On top of Spaghetti

3$ grep top oldsmokey /dev/null
oldsmokey:On top of Old Smoky
```

In a Perl regular expression, `\b` means “word boundary”: the beginning or end of a word that is not embedded in a larger word. Therefore you can say write the regular expression `\bsce\b` instead of `(^[^a-z_])sce([^a-z_]|$)`.

```
#!/bin/sh
#Output the lines that mention SCE in all the text files in my
#directories. Show the name of each file and the directory that
#contains it.

for directory in `find $HOME -type d -print`
do
    cd $directory
    echo
    echo $directory:

    for file in *
    do
        if file $file | grep -q text
        then
            grep -Ei 'Continuing Ed|(^|[^a-z_])sce([^a-z_]|$)' \
                $file /dev/null
        fi
    done
done

exit 0
```

The output of the shellscrip is:

```

/home/m/mmm64:
notes:wwwultra.sce.nyu.edu meretzky js1998
notes:wwwultra.sce.nyu.edu j$Cr98

/home/m/mmm64/public_html:
computer.html:<a href = "http://www.lab.sce.nyu.edu/wwwserver">web course</a>.
j.html:<a href="http://www.nyu.edu/sce">School of Continuing Education.</a>

/home/m/mmm64/public_html/x52.9264:
index.html:<a href = "news:nyu.sce.infotech.objc++">news:nyu.sce.infotech.objc++</a>

/home/m/mmm64/public_html/x52.9544:
index.html:<li><a href = "http://www.sce.nyu.edu/">School of Continuing Education</li>

```

The output of `find` is:

```

4$ find $HOME -type d -print | head -5
/home1/m/mmm64
/home1/m/mmm64/public_html
/home1/m/mmm64/public_html/fleisher
/home1/m/mmm64/public_html/goldfarb
/home1/m/mmm64/public_html/goldfarb/image

5$ file /etc/passwd
/etc/passwd: ascii text

7$ file /home1/m/mmm64/45/data/prog.c
/home1/m/mmm64/45/data/prog.c: cannot open: No such file or directory

8$ file /home1/m/mmm64/45/data/prog.s
/home1/m/mmm64/45/data/prog.s: assembler program text

9$ which date
/usr/ucb/date

10$ file /usr/ucb/date
/usr/ucb/date: cannot open: No such file or directory

11$ strings /usr/ucb/file | awk '465 <= NR && NR <= 477' | more
c program text
fortran program text
assembler program text
[nt]roff, tbl, or eqn input text
troff (CAT) output
commands text
troff intermediate output text
English text
ascii text

```

### Our three versions of awk

```
/bin/awk
/bin/nawk
/usr/xpg4/bin/awk
```

### What you can {print} with awk: pp. 114–130

(1) {print \$1}, etc., will print the specified field of each input line. **awk** will think the fields are separated by blanks or tabs unless you tell it otherwise with the uppercase **-F** option:

```
1$ date
```

```
Friday, January 25, 2008 12:17:42 PM EST
```

```
2$ date | awk '{print $4}'
```

```
2008
```

```
3$ date | awk '{print $4}' | awk -F: '{print $3}'
```

```
4$ more /etc/passwd
```

```
yqd6919:x:16033:15:Yevgeny Deek:/home1/y/yqd6919:/usr/local/etc/expiredshell
```

```
5$ awk -F: '{print $6}' /etc/passwd
```

```
/home1/y/yqd6919
```

```
6$ awk -F: '{print $6}' /etc/passwd | awk -F/ '{print $3}'
```

```
y
```

```
7$ awk -F: '{print $6}' /etc/passwd | grep '^/home/' | awk -F/ '{print $3}' | \
  sort | uniq -c | head -5
```

```
1 atar
1 basson
1 chapman
1 froese
1 itsdba
```

Later we'll rewrite the above example without **grep**:

```
8$ awk -F: '{print $6}' /etc/passwd | \
  awk -F/ 'NF >= 3 && $2 == "home" {print $3}' | \
  sort | uniq -c | head -5
```

(2) You can print any fields in any order. You can even repeat fields:

```
awk '{print $2, $1}'
awk '{print $2, $1, $1}'
```

```
#!/bin/ksh
#Write a backslash if the argument of awk is too long to fit.

awk '{print $2, $4, $6, $8, $10, $12, $14, $16, $18, $20, $22, $24, \
  $26}'
```

(3) \$0 means the whole input line (p. 116), so **awk '{print \$0, \$0}'** outputs two copies, side by side, of each line. {print} means {print \$0}.

(4) You can write expressions using fields or variables:

9\$ /usr/xpg4/bin/df -P

Filesystem	512-blocks	Used	Available	Capacity	Mounted on
/	16525738	4020822	12339660	25%	/
/dev	16525738	4020822	12339660	25%	/dev
/lib	20654264	9224974	11222748	46%	/lib
/local	20654264	9224974	11222748	46%	/local
/opt/SUNWspro	20654264	9224974	11222748	46%	/opt/SUNWspro
/opt/netbeans-5.5.1	20654264	9224974	11222748	46%	/opt/netbeans-5.5.1
/platform	20654264	9224974	11222748	46%	/platform
/sbin	20654264	9224974	11222748	46%	/sbin
/usr	20654264	9224974	11222748	46%	/usr
proc	0	0	0	0%	/proc
ctfs	0	0	0	0%	/system/contract
mnttab	0	0	0	0%	/etc/mnttab
objfs	0	0	0	0%	/system/object
swap	30205616	528	30205088	1%	/etc/svc/volatile
/platform/sun4u-us3/lib/libc_psr/libc_psr_hwcap1.so.1			20654264		9224974 112227
/platform/sun4u-us3/lib/sparcv9/libc_psr/libc_psr_hwcap1.so.1				20654264	9224974
fd	0	0	0	0%	/dev/fd
swap	30338464	133376	30205088	1%	/tmp
swap	30205152	64	30205088	1%	/var/run
i5pool/home1	140378112	56777586	67867499	46%	/home1
i5pool/home1/a	140378112	414	67867499	1%	/home1/a
i5pool/home1/b	140378112	183	67867499	1%	/home1/b
i5pool/home1/c	140378112	213	67867499	1%	/home1/c
i5pool/home1/d	140378112	213	67867499	1%	/home1/d
i5pool/home1/e	140378112	169	67867499	1%	/home1/e
i5pool/home1/f	140378112	69	67867499	1%	/home1/f
i5pool/home1/g	140378112	119	67867499	1%	/home1/g
i5pool/home1/h	140378112	111	67867499	1%	/home1/h
i5pool/home1/i	140378112	79	67867499	1%	/home1/i
i5pool/home1/j	140378112	436	67867499	1%	/home1/j
i5pool/home1/k	140378112	177	67867499	1%	/home1/k
i5pool/home1/l	140378112	119	67867499	1%	/home1/l
i5pool/home1/m	140378112	362	67867499	1%	/home1/m
i5pool/home1/n	140378112	161	67867499	1%	/home1/n
i5pool/home1/o	140378112	71	67867499	1%	/home1/o
i5pool/home1/p	140378112	127	67867499	1%	/home1/p
i5pool/home1/q	140378112	59	67867499	1%	/home1/q
i5pool/home1/r	140378112	207	67867499	1%	/home1/r
i5pool/home1/s	140378112	294	67867499	1%	/home1/s
i5pool/home1/t	140378112	157	67867499	1%	/home1/t
i5pool/home1/u	140378112	49	67867499	1%	/home1/u
i5pool/home1/v	140378112	111	67867499	1%	/home1/v
i5pool/home1/w	140378112	97	67867499	1%	/home1/w
i5pool/home1/x	140378112	63	67867499	1%	/home1/x
i5pool/home1/y	140378112	99	67867499	1%	/home1/y
i5pool/home1/z	140378112	67	67867499	1%	/home1/z
i5pool/home1/a/ab1258	204800	59	204741	1%	/home1/a/ab1258
i5pool/home1/a/ab215	204800	1225	203575	1%	/home1/a/ab215
i5pool/home1/a/abg237	204800	56	204744	1%	/home1/a/abg237
i5pool/home1/a/abs407	204800	56	204744	1%	/home1/a/abs407
i5pool/home1/a/ac1731	204800	57	204743	1%	/home1/a/ac1731
i5pool/home1/a/act285	204800	85763	119037	42%	/home1/a/act285

i5pool/home1/a/acw267	204800	10780	194020	6%	/home1/a/acw267
i5pool/home1/a/ad1180	204800	56	204744	1%	/home1/a/ad1180
i5pool/home1/a/ad35	204800	56	204744	1%	/home1/a/ad35
i5pool/home1/a/adb232	204800	4093	200707	2%	/home1/a/adb232
i5pool/home1/a/adj235	307200	47071	260129	16%	/home1/a/adj235
i5pool/home1/a/aec324	204800	4095	200705	2%	/home1/a/aec324
i5pool/home1/a/aeh310	204800	56	204744	1%	/home1/a/aeh310
i5pool/home1/a/afp233	204800	56	204744	1%	/home1/a/afp233
i5pool/home1/a/ag1	204800	102721	102079	51%	/home1/a/ag1
i5pool/home1/a/ag1564	204800	22128	182672	11%	/home1/a/ag1564
i5pool/home1/a/ag63	204800	42086	162714	21%	/home1/a/ag63
i5pool/home1/a/agh222	204800	49866	154934	25%	/home1/a/agh222
i5pool/home1/a/ahf3	204800	3509	201291	2%	/home1/a/ahf3
i5pool/home1/a/ah1257	204800	65	204735	1%	/home1/a/ah1257
i5pool/home1/a/ahs314	204800	56	204744	1%	/home1/a/ahs314
i5pool/home1/a/aja308	204800	56	204744	1%	/home1/a/aja308
i5pool/home1/a/ajb243	204800	44627	160173	22%	/home1/a/ajb243
i5pool/home1/a/ajg360	204800	10553	194247	6%	/home1/a/ajg360
i5pool/home1/a/ajg372	204800	56	204744	1%	/home1/a/ajg372
i5pool/home1/a/ajp236	204800	75	204725	1%	/home1/a/ajp236
i5pool/home1/a/ajs473	204800	56	204744	1%	/home1/a/ajs473
i5pool/home1/a/ak1726	204800	65	204735	1%	/home1/a/ak1726
i5pool/home1/a/ak213	204800	483	204317	1%	/home1/a/ak213
i5pool/home1/a/ak2445	204800	56	204744	1%	/home1/a/ak2445
i5pool/home1/a/ak60	204800	2596	202204	2%	/home1/a/ak60
i5pool/home1/a/akp258	204800	56	204744	1%	/home1/a/akp258
i5pool/home1/a/aks222	204800	426	204374	1%	/home1/a/aks222
i5pool/home1/a/al1345	204800	24864	179936	13%	/home1/a/al1345
i5pool/home1/a/al32	204800	56	204744	1%	/home1/a/al32
i5pool/home1/a/alb450	204800	56	204744	1%	/home1/a/alb450
i5pool/home1/a/als444	204800	77816	126984	38%	/home1/a/als444
i5pool/home1/a/am26	307200	201184	106016	66%	/home1/a/am26
i5pool/home1/a/am2694	204800	93329	111471	46%	/home1/a/am2694
i5pool/home1/a/am58	204800	56	204744	1%	/home1/a/am58
i5pool/home1/a/amb599	204800	56	204744	1%	/home1/a/amb599
i5pool/home1/a/amg516	204800	56	204744	1%	/home1/a/amg516
i5pool/home1/a/amg568	204800	101127	103673	50%	/home1/a/amg568
i5pool/home1/a/amk408	204800	25685	179115	13%	/home1/a/amk408
i5pool/home1/a/aml500	204800	56	204744	1%	/home1/a/aml500
i5pool/home1/a/ams5	204800	56	204744	1%	/home1/a/ams5
i5pool/home1/a/ana221	204800	6430	198370	4%	/home1/a/ana221
i5pool/home1/a/ap1512	204800	35006	169794	18%	/home1/a/ap1512
i5pool/home1/a/ap30	204800	56	204744	1%	/home1/a/ap30
i5pool/home1/a/apa2	204800	101971	102829	50%	/home1/a/apa2
i5pool/home1/a/aps237	204800	1238	203562	1%	/home1/a/aps237
i5pool/home1/a/ar93	204800	56	204744	1%	/home1/a/ar93
i5pool/home1/a/arr5400	204800	25227	179573	13%	/home1/a/arr5400
i5pool/home1/a/ars2	204800	56	204744	1%	/home1/a/ars2
i5pool/home1/a/as216	204800	40037	164763	20%	/home1/a/as216
i5pool/home1/a/asa200	204800	23653	181147	12%	/home1/a/asa200
i5pool/home1/a/asg311	204800	56	204744	1%	/home1/a/asg311
i5pool/home1/a/ash316	204800	56	204744	1%	/home1/a/ash316
i5pool/home1/a/asj237	204800	56	204744	1%	/home1/a/asj237
i5pool/home1/a/asl339	204800	56	204744	1%	/home1/a/asl339



i5pool/home1/a/att210	204800	5466	199334	3%	/home1/a/att210
i5pool/home1/a/aw987	204800	56	204744	1%	/home1/a/aw987
i5pool/home1/a/aya1	204800	56	204744	1%	/home1/a/aya1
i5pool/home1/b/ba4	204800	23788	181012	12%	/home1/b/ba4
i5pool/home1/b/bar286	204800	56	204744	1%	/home1/b/bar286
i5pool/home1/b/bc1211	204800	56	204744	1%	/home1/b/bc1211
i5pool/home1/b/bds231	204800	34349	170451	17%	/home1/b/bds231
i5pool/home1/b/bem2533	204800	56	204744	1%	/home1/b/bem2533
i5pool/home1/b/bg3	204800	56	204744	1%	/home1/b/bg3
i5pool/home1/b/bh1223	204800	56	204744	1%	/home1/b/bh1223
i5pool/home1/b/bj1271	204800	74359	130441	37%	/home1/b/bj1271
i5pool/home1/b/bjr270	204800	61632	143168	31%	/home1/b/bjr270
i5pool/home1/b/bk723	204800	25990	178810	13%	/home1/b/bk723
i5pool/home1/b/bl18	204800	56	204744	1%	/home1/b/bl18
i5pool/home1/b/blb267	204800	56	204744	1%	/home1/b/blb267
i5pool/home1/b/blr246	204800	57	204743	1%	/home1/b/blr246
i5pool/home1/b/bno1	204800	56	204744	1%	/home1/b/bno1
i5pool/home1/b/bo2	204800	56	204744	1%	/home1/b/bo2
i5pool/home1/b/bp439	204800	56	204744	1%	/home1/b/bp439
i5pool/home1/b/brh226	204800	72770	132030	36%	/home1/b/brh226
i5pool/home1/b/bv326	204800	539	204261	1%	/home1/b/bv326
i5pool/home1/b/bw609	204800	56	204744	1%	/home1/b/bw609
i5pool/home1/b/bws7157	204800	194014	10786	95%	/home1/b/bws7157
i5pool/home1/c/cab503	204800	56	204744	1%	/home1/c/cab503
i5pool/home1/c/cak5	204800	56	204744	1%	/home1/c/cak5
i5pool/home1/c/car265	204800	34937	169863	18%	/home1/c/car265
i5pool/home1/c/cas277	204800	20026	184774	10%	/home1/c/cas277
i5pool/home1/c/cb103	204800	235	204565	1%	/home1/c/cb103
i5pool/home1/c/cb2	204800	56	204744	1%	/home1/c/cb2
i5pool/home1/c/cb80	204800	78	204722	1%	/home1/c/cb80
i5pool/home1/c/cc73	204800	206	204594	1%	/home1/c/cc73
i5pool/home1/c/cea258	204800	13140	191660	7%	/home1/c/cea258
i5pool/home1/c/cf2	307200	217715	89485	71%	/home1/c/cf2
i5pool/home1/c/cgp225	204800	56	204744	1%	/home1/c/cgp225
i5pool/home1/c/ch679	204800	56	204744	1%	/home1/c/ch679
i5pool/home1/c/changj06	204800	1506	203294	1%	/home1/c/changj06
i5pool/home1/c/chapman	204800	35483	169317	18%	/home1/c/chapman
i5pool/home1/c/ciu200	204800	55161	149639	27%	/home1/c/ciu200
i5pool/home1/c/ck1047	204800	52404	152396	26%	/home1/c/ck1047
i5pool/home1/c/ckg214	204800	56	204744	1%	/home1/c/ckg214
i5pool/home1/c/ckw200	204800	17419	187381	9%	/home1/c/ckw200
i5pool/home1/c/clc323	204800	57	204743	1%	/home1/c/clc323
i5pool/home1/c/cm1708	204800	2833	201967	2%	/home1/c/cm1708
i5pool/home1/c/cm676	204800	92089	112711	45%	/home1/c/cm676
i5pool/home1/c/cmh1	204800	56	204744	1%	/home1/c/cmh1
i5pool/home1/c/cml371	204800	100136	104664	49%	/home1/c/cml371
i5pool/home1/c/cmm533	204800	48108	156692	24%	/home1/c/cmm533
i5pool/home1/c/cmp288	204800	28713	176087	15%	/home1/c/cmp288
i5pool/home1/c/cn341	204800	39104	165696	20%	/home1/c/cn341
i5pool/home1/c/cp510	204800	13783	191017	7%	/home1/c/cp510
i5pool/home1/c/cpm281	204800	57	204743	1%	/home1/c/cpm281
i5pool/home1/c/crs297	204800	45680	159120	23%	/home1/c/crs297
i5pool/home1/c/csa219	204800	56	204744	1%	/home1/c/csa219
i5pool/home1/c/cwc271	204800	56	204744	1%	/home1/c/cwc271

i5pool/home1/c/cwt1	204800	25130	179670	13%	/home1/c/cwt1
i5pool/home1/c/cwy202	204800	56	204744	1%	/home1/c/cwy202
i5pool/home1/c/cy1	204800	60	204740	1%	/home1/c/cy1
i5pool/home1/d/dac3	204800	56	204744	1%	/home1/d/dac3
i5pool/home1/d/dac5716	204800	56	204744	1%	/home1/d/dac5716
i5pool/home1/d/dap240	204800	28751	176049	15%	/home1/d/dap240
i5pool/home1/d/das551	204800	56	204744	1%	/home1/d/das551
i5pool/home1/d/db13	204800	18038	186762	9%	/home1/d/db13
i5pool/home1/d/db225	204800	56	204744	1%	/home1/d/db225
i5pool/home1/d/dbb212	409600	169225	240375	42%	/home1/d/dbb212
i5pool/home1/d/dbftest	204800	1975	202825	1%	/home1/d/dbftest
i5pool/home1/d/dbs295	204800	56	204744	1%	/home1/d/dbs295
i5pool/home1/d/dd9	204800	56	204744	1%	/home1/d/dd9
i5pool/home1/d/deb298	204800	24216	180584	12%	/home1/d/deb298
i5pool/home1/d/df3	204800	56	204744	1%	/home1/d/df3
i5pool/home1/d/df779	204800	56	204744	1%	/home1/d/df779
i5pool/home1/d/dfr229	204800	56	204744	1%	/home1/d/dfr229
i5pool/home1/d/dfv200	204800	305	204495	1%	/home1/d/dfv200
i5pool/home1/d/dg60	204800	71	204729	1%	/home1/d/dg60
i5pool/home1/d/djb259	204800	29918	174882	15%	/home1/d/djb259
i5pool/home1/d/djc391	204800	56	204744	1%	/home1/d/djc391
i5pool/home1/d/dkd210	204800	56	204744	1%	/home1/d/dkd210
i5pool/home1/d/dm129	204800	17068	187732	9%	/home1/d/dm129
i5pool/home1/d/dm733	204800	1234	203566	1%	/home1/d/dm733
i5pool/home1/d/dm757	204800	56	204744	1%	/home1/d/dm757
i5pool/home1/d/dma306	204800	56	204744	1%	/home1/d/dma306
i5pool/home1/d/dp1006	204800	56	204744	1%	/home1/d/dp1006
i5pool/home1/d/dpc224	204800	56	204744	1%	/home1/d/dpc224
i5pool/home1/d/dqb3240	204800	10687	194113	6%	/home1/d/dqb3240
i5pool/home1/d/drf7136	204800	4683	200117	3%	/home1/d/drf7136
i5pool/home1/d/ds905	204800	40587	164213	20%	/home1/d/ds905
i5pool/home1/d/dsb1	204800	174812	29988	86%	/home1/d/dsb1
i5pool/home1/d/dse7916	204800	25730	179070	13%	/home1/d/dse7916
i5pool/home1/d/dtk4926	204800	56	204744	1%	/home1/d/dtk4926
i5pool/home1/d/dv441	204800	33943	170857	17%	/home1/d/dv441
i5pool/home1/d/dvg209	204800	56	204744	1%	/home1/d/dvg209
i5pool/home1/d/dwk240	204800	60	204740	1%	/home1/d/dwk240
i5pool/home1/e/eag235	204800	8566	196234	5%	/home1/e/eag235
i5pool/home1/e/eal216	204800	30501	174299	15%	/home1/e/eal216
i5pool/home1/e/ear1	204800	7369	197431	4%	/home1/e/ear1
i5pool/home1/e/eb7	204800	56	204744	1%	/home1/e/eb7
i5pool/home1/e/ec1098	204800	56	204744	1%	/home1/e/ec1098
i5pool/home1/e/ed1	204800	2508	202292	2%	/home1/e/ed1
i5pool/home1/e/ed30	204800	18556	186244	10%	/home1/e/ed30
i5pool/home1/e/ees276	204800	27539	177261	14%	/home1/e/ees276
i5pool/home1/e/ehk246	204800	20540	184260	11%	/home1/e/ehk246
i5pool/home1/e/ei11	204800	93	204707	1%	/home1/e/ei11
i5pool/home1/e/ek61	204800	2573	202227	2%	/home1/e/ek61
i5pool/home1/e/ek17581	204800	16404	188396	9%	/home1/e/ek17581
i5pool/home1/e/els4	204800	2812	201988	2%	/home1/e/els4
i5pool/home1/e/em15	204800	56	204744	1%	/home1/e/em15
i5pool/home1/e/emd0753	204800	1947	202853	1%	/home1/e/emd0753
i5pool/home1/e/emf282	204800	1765	203035	1%	/home1/e/emf282
i5pool/home1/e/emk1120	204800	401	204399	1%	/home1/e/emk1120

i5pool/home1/e/ep397	204800	56	204744	1%	/home1/e/ep397
i5pool/home1/e/er929	204800	57	204743	1%	/home1/e/er929
i5pool/home1/e/erw234	307200	22776	284424	8%	/home1/e/erw234
i5pool/home1/e/es106	204800	149	204651	1%	/home1/e/es106
i5pool/home1/e/esk274	204800	56	204744	1%	/home1/e/esk274
i5pool/home1/e/et662	204800	56	204744	1%	/home1/e/et662
i5pool/home1/e/eyl219	204800	272	204528	1%	/home1/e/eyl219
i5pool/home1/e/ez6	204800	1382	203418	1%	/home1/e/ez6
i5pool/home1/f/far238	204800	56	204744	1%	/home1/f/far238
i5pool/home1/f/fbb1	204800	133	204667	1%	/home1/f/fbb1
i5pool/home1/f/feg1	204800	300	204500	1%	/home1/f/feg1
i5pool/home1/f/frd206	204800	56	204744	1%	/home1/f/frd206
i5pool/home1/g/gaf246	204800	57877	146923	29%	/home1/g/gaf246
i5pool/home1/g/gag260	204800	90185	114615	45%	/home1/g/gag260
i5pool/home1/g/gaj232	204800	56	204744	1%	/home1/g/gaj232
i5pool/home1/g/gb886	204800	56	204744	1%	/home1/g/gb886
i5pool/home1/g/gd586	204800	94421	110379	47%	/home1/g/gd586
i5pool/home1/g/gfm1	204800	286	204514	1%	/home1/g/gfm1
i5pool/home1/g/gh403	204800	152	204648	1%	/home1/g/gh403
i5pool/home1/g/gm1011	204800	56	204744	1%	/home1/g/gm1011
i5pool/home1/g/gma240	204800	15275	189525	8%	/home1/g/gma240
i5pool/home1/g/gp204	204800	56	204744	1%	/home1/g/gp204
i5pool/home1/g/gqv1359	204800	20976	183824	11%	/home1/g/gqv1359
i5pool/home1/g/gs491	204800	1583	203217	1%	/home1/g/gs491
i5pool/home1/g/gwc1	204800	57	204743	1%	/home1/g/gwc1
i5pool/home1/h/hag226	204800	56	204744	1%	/home1/h/hag226
i5pool/home1/h/hartj01	204800	56	204744	1%	/home1/h/hartj01
i5pool/home1/h/hd27	204800	556	204244	1%	/home1/h/hd27
i5pool/home1/h/hf1	204800	29600	175200	15%	/home1/h/hf1
i5pool/home1/h/hjf223	204800	5211	199589	3%	/home1/h/hjf223
i5pool/home1/h/hk218	204800	56	204744	1%	/home1/h/hk218
i5pool/home1/h/hka203	204800	56	204744	1%	/home1/h/hka203
i5pool/home1/h/hkw204	204800	56	204744	1%	/home1/h/hkw204
i5pool/home1/h/hms269	307200	67763	239437	23%	/home1/h/hms269
i5pool/home1/h/hnp6713	204800	456	204344	1%	/home1/h/hnp6713
i5pool/home1/h/hsp223	204800	1040	203760	1%	/home1/h/hsp223
i5pool/home1/h/hwb210	204800	66737	138063	33%	/home1/h/hwb210
i5pool/home1/i/iaf203	204800	59	204741	1%	/home1/i/iaf203
i5pool/home1/i/id2	204800	56	204744	1%	/home1/i/id2
i5pool/home1/i/id251	204800	27864	176936	14%	/home1/i/id251
i5pool/home1/i/iis200	204800	102	204698	1%	/home1/i/iis200
i5pool/home1/i/im36	204800	59	204741	1%	/home1/i/im36
i5pool/home1/i/itsdba	204800	5956	198844	3%	/home1/i/itsdba
i5pool/home1/i/izb6225	204800	26180	178620	13%	/home1/i/izb6225
i5pool/home1/j/jag1	204800	203	204597	1%	/home1/j/jag1
i5pool/home1/j/jag548	204800	56	204744	1%	/home1/j/jag548
i5pool/home1/j/jai223	204800	56	204744	1%	/home1/j/jai223
i5pool/home1/j/jak5680	204800	56	204744	1%	/home1/j/jak5680
i5pool/home1/j/jan225	204800	5722	199078	3%	/home1/j/jan225
i5pool/home1/j/jao247	204800	12510	192290	7%	/home1/j/jao247
i5pool/home1/j/jao294	307200	128050	179150	42%	/home1/j/jao294
i5pool/home1/j/jb175	204800	75	204725	1%	/home1/j/jb175
i5pool/home1/j/jb4	204800	175128	29672	86%	/home1/j/jb4
i5pool/home1/j/jbf284	204800	56	204744	1%	/home1/j/jbf284

i5pool/home1/j/jc2942	204800	56	204744	1%	/home1/j/jc2942
i5pool/home1/j/jc73	204800	184	204616	1%	/home1/j/jc73
i5pool/home1/j/jcf2	307200	252434	54766	83%	/home1/j/jcf2
i5pool/home1/j/jcs296	204800	16777	188023	9%	/home1/j/jcs296
i5pool/home1/j/jcw316	204800	56	204744	1%	/home1/j/jcw316
i5pool/home1/j/jd1442	204800	18498	186302	10%	/home1/j/jd1442
i5pool/home1/j/jda299	204800	56	204744	1%	/home1/j/jda299
i5pool/home1/j/jdc7	204800	44130	160670	22%	/home1/j/jdc7
i5pool/home1/j/jeg331	204800	27908	176892	14%	/home1/j/jeg331
i5pool/home1/j/jek339	204800	56	204744	1%	/home1/j/jek339
i5pool/home1/j/jfs321	204800	56	204744	1%	/home1/j/jfs321
i5pool/home1/j/jg1636	204800	412	204388	1%	/home1/j/jg1636
i5pool/home1/j/jg15296	204800	111	204689	1%	/home1/j/jg15296
i5pool/home1/j/jh2	204800	57	204743	1%	/home1/j/jh2
i5pool/home1/j/jhh290	204800	56	204744	1%	/home1/j/jhh290
i5pool/home1/j/jhs227	204800	10295	194505	6%	/home1/j/jhs227
i5pool/home1/j/jjg224	204800	56	204744	1%	/home1/j/jjg224
i5pool/home1/j/jj12	204800	56	204744	1%	/home1/j/jj12
i5pool/home1/j/jjt272	204800	19803	184997	10%	/home1/j/jjt272
i5pool/home1/j/jjx203	204800	56	204744	1%	/home1/j/jjx203
i5pool/home1/j/jjy225	204800	57	204743	1%	/home1/j/jjy225
i5pool/home1/j/jk1772	204800	56	204744	1%	/home1/j/jk1772
i5pool/home1/j/jkb1	204800	56	204744	1%	/home1/j/jkb1
i5pool/home1/j/jkh210	204800	56	204744	1%	/home1/j/jkh210
i5pool/home1/j/jl11	204800	908	203892	1%	/home1/j/jl11
i5pool/home1/j/jl1163	204800	25857	178943	13%	/home1/j/jl1163
i5pool/home1/j/jl2462	204800	29554	175246	15%	/home1/j/jl2462
i5pool/home1/j/jl3	204800	1276	203524	1%	/home1/j/jl3
i5pool/home1/j/jl4	204800	225	204575	1%	/home1/j/jl4
i5pool/home1/j/jl531	204800	179	204621	1%	/home1/j/jl531
i5pool/home1/j/jld1	204800	60045	144755	30%	/home1/j/jld1
i5pool/home1/j/jlh393	204800	59	204741	1%	/home1/j/jlh393
i5pool/home1/j/jl1387	204800	56	204744	1%	/home1/j/jl1387
i5pool/home1/j/jlr349	204800	51288	153512	26%	/home1/j/jlr349
i5pool/home1/j/jm1396	204800	818	203982	1%	/home1/j/jm1396
i5pool/home1/j/jm1543	204800	23451	181349	12%	/home1/j/jm1543
i5pool/home1/j/jmb657	204800	56	204744	1%	/home1/j/jmb657
i5pool/home1/j/jmm636	204800	56	204744	1%	/home1/j/jmm636
i5pool/home1/j/jmr363	204800	331	204469	1%	/home1/j/jmr363
i5pool/home1/j/jms239	204800	56	204744	1%	/home1/j/jms239
i5pool/home1/j/jms755	204800	37300	167500	19%	/home1/j/jms755
i5pool/home1/j/jos5015	204800	1848	202952	1%	/home1/j/jos5015
i5pool/home1/j/jqc1	204800	59	204741	1%	/home1/j/jqc1
i5pool/home1/j/jr42	204800	56	204744	1%	/home1/j/jr42
i5pool/home1/j/jr669	204800	1246	203554	1%	/home1/j/jr669
i5pool/home1/j/jrc239	204800	1576	203224	1%	/home1/j/jrc239
i5pool/home1/j/jrc260	204800	6517	198283	4%	/home1/j/jrc260
i5pool/home1/j/jrn218	204800	1327	203473	1%	/home1/j/jrn218
i5pool/home1/j/jsb3	665600	632080	33520	95%	/home1/j/jsb3
i5pool/home1/j/jsh382	204800	56	204744	1%	/home1/j/jsh382
i5pool/home1/j/jss1	204800	244	204556	1%	/home1/j/jss1
i5pool/home1/j/jss279	204800	9082	195718	5%	/home1/j/jss279
i5pool/home1/j/jvd211	204800	1062	203738	1%	/home1/j/jvd211
i5pool/home1/j/jwg263	204800	56	204744	1%	/home1/j/jwg263

i5pool/home1/j/jy594	204800	56	204744	1%	/home1/j/jy594
i5pool/home1/k/kaa280	204800	9721	195079	5%	/home1/k/kaa280
i5pool/home1/k/kc1149	204800	56	204744	1%	/home1/k/kc1149
i5pool/home1/k/kck1	204800	176	204624	1%	/home1/k/kck1
i5pool/home1/k/kd722	204800	12279	192521	6%	/home1/k/kd722
i5pool/home1/k/kf632	204800	56	204744	1%	/home1/k/kf632
i5pool/home1/k/ki1207	204800	56	204744	1%	/home1/k/ki1207
i5pool/home1/k/kk48	204800	56	204744	1%	/home1/k/kk48
i5pool/home1/k/kkg209	204800	27307	177493	14%	/home1/k/kkg209
i5pool/home1/k/kkw235	204800	56	204744	1%	/home1/k/kkw235
i5pool/home1/k/klk291	204800	171326	33474	84%	/home1/k/klk291
i5pool/home1/k/km602	204800	863	203937	1%	/home1/k/km602
i5pool/home1/k/kmw323	204800	56	204744	1%	/home1/k/kmw323
i5pool/home1/k/kos6259	204800	35133	169667	18%	/home1/k/kos6259
i5pool/home1/k/kp1	204800	56	204744	1%	/home1/k/kp1
i5pool/home1/k/kpd200	204800	241	204559	1%	/home1/k/kpd200
i5pool/home1/k/kqg6183	204800	56	204744	1%	/home1/k/kqg6183
i5pool/home1/k/ktj208	204800	2611	202189	2%	/home1/k/ktj208
i5pool/home1/k/kw366	204800	1755	203045	1%	/home1/k/kw366
i5pool/home1/l/las560	204800	56	204744	1%	/home1/l/las560
i5pool/home1/l/lb99	204800	265	204535	1%	/home1/l/lb99
i5pool/home1/l/lbh246	204800	56	204744	1%	/home1/l/lbh246
i5pool/home1/l/ld456	204800	39210	165590	20%	/home1/l/ld456
i5pool/home1/l/lds287	204800	67	204733	1%	/home1/l/lds287
i5pool/home1/l/ldt224	204800	56720	148080	28%	/home1/l/ldt224
i5pool/home1/l/limb03	204800	7608	197192	4%	/home1/l/limb03
i5pool/home1/l/l1337	204800	30368	174432	15%	/home1/l/l1337
i5pool/home1/l/l1470	204800	17728	187072	9%	/home1/l/l1470
i5pool/home1/l/lml353	204800	13066	191734	7%	/home1/l/lml353
i5pool/home1/l/lmr291	204800	100272	104528	49%	/home1/l/lmr291
i5pool/home1/l/lpm217	204800	21901	182899	11%	/home1/l/lpm217
i5pool/home1/l/l1t680	204800	56	204744	1%	/home1/l/l1t680
i5pool/home1/l/lxn202	204800	147	204653	1%	/home1/l/lxn202
i5pool/home1/m/maa237	204800	1762	203038	1%	/home1/m/maa237
i5pool/home1/m/mab579	204800	56	204744	1%	/home1/m/mab579
i5pool/home1/m/mac656	204800	56	204744	1%	/home1/m/mac656
i5pool/home1/m/maf2085	204800	56	204744	1%	/home1/m/maf2085
i5pool/home1/m/mah533	204800	56	204744	1%	/home1/m/mah533
i5pool/home1/m/mak413	204800	56	204744	1%	/home1/m/mak413
i5pool/home1/m/mar297	204800	64000	140800	32%	/home1/m/mar297
i5pool/home1/m/mas507	204800	56	204744	1%	/home1/m/mas507
i5pool/home1/m/mas789	204800	56	204744	1%	/home1/m/mas789
i5pool/home1/m/mb3	204800	56	204744	1%	/home1/m/mb3
i5pool/home1/m/mb3093	204800	56	204744	1%	/home1/m/mb3093
i5pool/home1/m/mb484	204800	19915	184885	10%	/home1/m/mb484
i5pool/home1/m/mbs348	204800	56	204744	1%	/home1/m/mbs348
i5pool/home1/m/mc159	204800	822	203978	1%	/home1/m/mc159
i5pool/home1/m/mcp249	204800	13042	191758	7%	/home1/m/mcp249
i5pool/home1/m/md200	204800	56	204744	1%	/home1/m/md200
i5pool/home1/m/mdm410	204800	56	204744	1%	/home1/m/mdm410
i5pool/home1/m/mds224	204800	248	204552	1%	/home1/m/mds224
i5pool/home1/m/mec9411	204800	56	204744	1%	/home1/m/mec9411
i5pool/home1/m/mel6	204800	57	204743	1%	/home1/m/mel6
i5pool/home1/m/mf1207	204800	56	204744	1%	/home1/m/mf1207

i5pool/home1/m/mfv215	204800	14308	190492	7%	/home1/m/mfv215
i5pool/home1/m/mga250	204800	56	204744	1%	/home1/m/mga250
i5pool/home1/m/mhl219	204800	56	204744	1%	/home1/m/mhl219
i5pool/home1/m/miy201	204800	56	204744	1%	/home1/m/miy201
i5pool/home1/m/mjl1	204800	37544	167256	19%	/home1/m/mjl1
i5pool/home1/m/mjp201	204800	56	204744	1%	/home1/m/mjp201
i5pool/home1/m/mjs325	204800	23862	180938	12%	/home1/m/mjs325
i5pool/home1/m/mk125	204800	60	204740	1%	/home1/m/mk125
i5pool/home1/m/mk2627	204800	56	204744	1%	/home1/m/mk2627
i5pool/home1/m/mks266	204800	56	204744	1%	/home1/m/mks266
i5pool/home1/m/ml50	204800	56	204744	1%	/home1/m/ml50
i5pool/home1/m/ml8	204800	42224	162576	21%	/home1/m/ml8
i5pool/home1/m/mlc413	204800	56	204744	1%	/home1/m/mlc413
i5pool/home1/m/mm3572	204800	487	204313	1%	/home1/m/mm3572
i5pool/home1/m/mm3692	204800	56	204744	1%	/home1/m/mm3692
i5pool/home1/m/mm4180	204800	50086	154714	25%	/home1/m/mm4180
i5pool/home1/m/mm64	1024000	615796	408204	61%	/home1/m/mm64
i5pool/home1/m/mo21	204800	598	204202	1%	/home1/m/mo21
i5pool/home1/m/mp1	409600	45542	364058	12%	/home1/m/mp1
i5pool/home1/m/mp1707	204800	8488	196312	5%	/home1/m/mp1707
i5pool/home1/m/mp26	204800	56	204744	1%	/home1/m/mp26
i5pool/home1/m/mp35	204800	56	204744	1%	/home1/m/mp35
i5pool/home1/m/mq268	204800	56	204744	1%	/home1/m/mq268
i5pool/home1/m/mqr5025	204800	201816	2984	99%	/home1/m/mqr5025
i5pool/home1/m/mr24	204800	219	204581	1%	/home1/m/mr24
i5pool/home1/m/mrc229	204800	31165	173635	16%	/home1/m/mrc229
i5pool/home1/m/mrk294	204800	56	204744	1%	/home1/m/mrk294
i5pool/home1/m/mrs426	204800	56	204744	1%	/home1/m/mrs426
i5pool/home1/m/ms126	204800	40828	163972	20%	/home1/m/ms126
i5pool/home1/m/ms178	204800	118	204682	1%	/home1/m/ms178
i5pool/home1/m/ms312	204800	56	204744	1%	/home1/m/ms312
i5pool/home1/m/msk285	204800	56	204744	1%	/home1/m/msk285
i5pool/home1/m/msl334	204800	136628	68172	67%	/home1/m/msl334
i5pool/home1/m/msm311	204800	101	204699	1%	/home1/m/msm311
i5pool/home1/m/mt60	204800	56	204744	1%	/home1/m/mt60
i5pool/home1/m/mu248	204800	56	204744	1%	/home1/m/mu248
i5pool/home1/m/mwl1119	204800	1085	203715	1%	/home1/m/mwl1119
i5pool/home1/m/mww2	204800	137	204663	1%	/home1/m/mww2
i5pool/home1/m/mxp202	204800	56	204744	1%	/home1/m/mxp202
i5pool/home1/m/mysql	204800	56	204744	1%	/home1/m/mysql
i5pool/home1/m/mzs212	204800	56	204744	1%	/home1/m/mzs212
i5pool/home1/n/na426	204800	56	204744	1%	/home1/n/na426
i5pool/home1/n/na563	204800	44389	160411	22%	/home1/n/na563
i5pool/home1/n/nar2006	204800	56	204744	1%	/home1/n/nar2006
i5pool/home1/n/nc771	204800	56	204744	1%	/home1/n/nc771
i5pool/home1/n/ncw213	204800	4423	200377	3%	/home1/n/ncw213
i5pool/home1/n/nds229	204800	5744	199056	3%	/home1/n/nds229
i5pool/home1/n/ng13	204800	56	204744	1%	/home1/n/ng13
i5pool/home1/n/nh2	1024000	417200	606800	41%	/home1/n/nh2
i5pool/home1/n/nh416	204800	41583	163217	21%	/home1/n/nh416
i5pool/home1/n/nkd211	204800	56	204744	1%	/home1/n/nkd211
i5pool/home1/n/nl29	204800	175	204625	1%	/home1/n/nl29
i5pool/home1/n/nlb1	204800	2249	202551	2%	/home1/n/nlb1
i5pool/home1/n/nml	204800	209	204591	1%	/home1/n/nml

i5pool/home1/n/nmd254	204800	56	204744	1%	/home1/n/nmd254
i5pool/home1/n/nrb231	204800	62	204738	1%	/home1/n/nrb231
i5pool/home1/n/nsd2	204800	56	204744	1%	/home1/n/nsd2
i5pool/home1/n/nsn210	204800	56	204744	1%	/home1/n/nsn210
i5pool/home1/n/nvg200	204800	56	204744	1%	/home1/n/nvg200
i5pool/home1/n/nyu28	204800	117	204683	1%	/home1/n/nyu28
i5pool/home1/o/oan203	204800	125	204675	1%	/home1/o/oan203
i5pool/home1/o/obw1	204800	56	204744	1%	/home1/o/obw1
i5pool/home1/o/ohg200	204800	49880	154920	25%	/home1/o/ohg200
i5pool/home1/o/ohv200	204800	56	204744	1%	/home1/o/ohv200
i5pool/home1/o/ojc201	204800	31835	172965	16%	/home1/o/ojc201
i5pool/home1/o/om273	204800	58	204742	1%	/home1/o/om273
i5pool/home1/p/pa29	204800	26397	178403	13%	/home1/p/pa29
i5pool/home1/p/pa39	204800	56	204744	1%	/home1/p/pa39
i5pool/home1/p/pb923	204800	132344	72456	65%	/home1/p/pb923
i5pool/home1/p/pc468	204800	18378	186422	9%	/home1/p/pc468
i5pool/home1/p/pc897	204800	38584	166216	19%	/home1/p/pc897
i5pool/home1/p/pc965	204800	32466	172334	16%	/home1/p/pc965
i5pool/home1/p/ph10	204800	56	204744	1%	/home1/p/ph10
i5pool/home1/p/pjr258	204800	56	204744	1%	/home1/p/pjr258
i5pool/home1/p/pln217	409600	327172	82428	80%	/home1/p/pln217
i5pool/home1/p/pmc316	204800	56	204744	1%	/home1/p/pmc316
i5pool/home1/p/pmf239	204800	41356	163444	21%	/home1/p/pmf239
i5pool/home1/p/pp720	204800	56	204744	1%	/home1/p/pp720
i5pool/home1/p/pp747	204800	56	204744	1%	/home1/p/pp747
i5pool/home1/p/pr656	204800	65	204735	1%	/home1/p/pr656
i5pool/home1/p/ps6	204800	56	204744	1%	/home1/p/ps6
i5pool/home1/p/pv248	204800	2479	202321	2%	/home1/p/pv248
i5pool/home1/p/pw1	204800	61	204739	1%	/home1/p/pw1
i5pool/home1/q/ql212	204800	56	204744	1%	/home1/q/ql212
i5pool/home1/q/qlc201	204800	126102	78698	62%	/home1/q/qlc201
i5pool/home1/r/rag339	204800	402	204398	1%	/home1/r/rag339
i5pool/home1/r/ras311	204800	226	204574	1%	/home1/r/ras311
i5pool/home1/r/rau206	204800	56	204744	1%	/home1/r/rau206
i5pool/home1/r/rb72	204800	15906	188894	8%	/home1/r/rb72
i5pool/home1/r/rcd2	204800	56	204744	1%	/home1/r/rcd2
i5pool/home1/r/rd578	204800	28789	176011	15%	/home1/r/rd578
i5pool/home1/r/rdh268	204800	56	204744	1%	/home1/r/rdh268
i5pool/home1/r/reb200	204800	97	204703	1%	/home1/r/reb200
i5pool/home1/r/rew213	204800	31165	173635	16%	/home1/r/rew213
i5pool/home1/r/rg5	204800	59	204741	1%	/home1/r/rg5
i5pool/home1/r/rgg233	204800	33957	170843	17%	/home1/r/rgg233
i5pool/home1/r/rgp3	204800	332	204468	1%	/home1/r/rgp3
i5pool/home1/r/rh1115	204800	83108	121692	41%	/home1/r/rh1115
i5pool/home1/r/rj308	204800	17358	187442	9%	/home1/r/rj308
i5pool/home1/r/rj488	204800	63316	141484	31%	/home1/r/rj488
i5pool/home1/r/rjy211	204800	56	204744	1%	/home1/r/rjy211
i5pool/home1/r/rk1213	204800	56	204744	1%	/home1/r/rk1213
i5pool/home1/r/r11064	204800	45897	158903	23%	/home1/r/r11064
i5pool/home1/r/r1965	204800	56	204744	1%	/home1/r/r1965
i5pool/home1/r/rlh1	204800	56	204744	1%	/home1/r/rlh1
i5pool/home1/r/rln221	204800	56	204744	1%	/home1/r/rln221
i5pool/home1/r/rm2	204800	159	204641	1%	/home1/r/rm2
i5pool/home1/r/rma224	204800	56	204744	1%	/home1/r/rma224

i5pool/home1/r/rmt267	204800	56	204744	1%	/home1/r/rmt267
i5pool/home1/r/rnk232	204800	56	204744	1%	/home1/r/rnk232
i5pool/home1/r/rob1	204800	37288	167512	19%	/home1/r/rob1
i5pool/home1/r/rp466	204800	3207	201593	2%	/home1/r/rp466
i5pool/home1/r/rpz204	204800	56	204744	1%	/home1/r/rpz204
i5pool/home1/r/rr25	204800	56	204744	1%	/home1/r/rr25
i5pool/home1/r/rs4	204800	56	204744	1%	/home1/r/rs4
i5pool/home1/r/rsp255	204800	103	204697	1%	/home1/r/rsp255
i5pool/home1/s/sa1101	204800	56	204744	1%	/home1/s/sa1101
i5pool/home1/s/sa1206	204800	114251	90549	56%	/home1/s/sa1206
i5pool/home1/s/sa46	204800	40596	164204	20%	/home1/s/sa46
i5pool/home1/s/sao1	204800	157843	46957	78%	/home1/s/sao1
i5pool/home1/s/sas574	204800	56	204744	1%	/home1/s/sas574
i5pool/home1/s/sat298	204800	56	204744	1%	/home1/s/sat298
i5pool/home1/s/sb144	204800	936	203864	1%	/home1/s/sb144
i5pool/home1/s/sb239	204800	1907	202893	1%	/home1/s/sb239
i5pool/home1/s/scc3	204800	6801	197999	4%	/home1/s/scc3
i5pool/home1/s/sch283	204800	56	204744	1%	/home1/s/sch283
i5pool/home1/s/scr259	204800	58334	146466	29%	/home1/s/scr259
i5pool/home1/s/sd1229	204800	56	204744	1%	/home1/s/sd1229
i5pool/home1/s/sdd246	204800	56	204744	1%	/home1/s/sdd246
i5pool/home1/s/sdm314	204800	56	204744	1%	/home1/s/sdm314
i5pool/home1/s/sf43	204800	24251	180549	12%	/home1/s/sf43
i5pool/home1/s/sgt223	204800	56	204744	1%	/home1/s/sgt223
i5pool/home1/s/sh1249	204800	21381	183419	11%	/home1/s/sh1249
i5pool/home1/s/sh1323	204800	56	204744	1%	/home1/s/sh1323
i5pool/home1/s/sj30	204800	65	204735	1%	/home1/s/sj30
i5pool/home1/s/sjh357	204800	56	204744	1%	/home1/s/sjh357
i5pool/home1/s/slm1	204800	56	204744	1%	/home1/s/slm1
i5pool/home1/s/sma1	204800	220	204580	1%	/home1/s/sma1
i5pool/home1/s/smb8818	204800	56	204744	1%	/home1/s/smb8818
i5pool/home1/s/smc449	204800	77326	127474	38%	/home1/s/smc449
i5pool/home1/s/smy223	204800	1181	203619	1%	/home1/s/smy223
i5pool/home1/s/snn206	204800	56	204744	1%	/home1/s/snn206
i5pool/home1/s/so4	204800	56	204744	1%	/home1/s/so4
i5pool/home1/s/so554	204800	63	204737	1%	/home1/s/so554
i5pool/home1/s/sp10	204800	56	204744	1%	/home1/s/sp10
i5pool/home1/s/spg240	204800	56	204744	1%	/home1/s/spg240
i5pool/home1/s/sq268	204800	4427	200373	3%	/home1/s/sq268
i5pool/home1/s/sqs8467	204800	168	204632	1%	/home1/s/sqs8467
i5pool/home1/s/sr1512	204800	56	204744	1%	/home1/s/sr1512
i5pool/home1/s/sr99	204800	56	204744	1%	/home1/s/sr99
i5pool/home1/s/srp311	204800	56	204744	1%	/home1/s/srp311
i5pool/home1/s/ss1729	204800	56	204744	1%	/home1/s/ss1729
i5pool/home1/s/ss28	204800	56	204744	1%	/home1/s/ss28
i5pool/home1/s/ss3987	204800	56	204744	1%	/home1/s/ss3987
i5pool/home1/s/ssa232	204800	56	204744	1%	/home1/s/ssa232
i5pool/home1/s/ssz211	204800	48408	156392	24%	/home1/s/ssz211
i5pool/home1/s/st1043	204800	371	204429	1%	/home1/s/st1043
i5pool/home1/s/swm223	204800	2735	202065	2%	/home1/s/swm223
i5pool/home1/s/sy595	204800	85714	119086	42%	/home1/s/sy595
i5pool/home1/t/ta19	204800	56	204744	1%	/home1/t/ta19
i5pool/home1/t/taf222	204800	40669	164131	20%	/home1/t/taf222
i5pool/home1/t/tc855	204800	56	204744	1%	/home1/t/tc855



i5pool/home1/t/tfc2	204800	5491	199309	3%	/home1/t/tfc2
i5pool/home1/t/tk499	204800	939	203861	1%	/home1/t/tk499
i5pool/home1/t/tkc233	204800	62693	142107	31%	/home1/t/tkc233
i5pool/home1/t/tlh249	204800	56	204744	1%	/home1/t/tlh249
i5pool/home1/t/tmw217	204800	39942	164858	20%	/home1/t/tmw217
i5pool/home1/t/tn18	204800	56	204744	1%	/home1/t/tn18
i5pool/home1/t/tp1210	204800	56	204744	1%	/home1/t/tp1210
i5pool/home1/t/tqy200	204800	61	204739	1%	/home1/t/tqy200
i5pool/home1/t/tr224	204800	42345	162455	21%	/home1/t/tr224
i5pool/home1/t/trr239	204800	56	204744	1%	/home1/t/trr239
i5pool/home1/t/ts65	204800	59	204741	1%	/home1/t/ts65
i5pool/home1/t/tss260	204800	56	204744	1%	/home1/t/tss260
i5pool/home1/t/tst102	204800	60	204740	1%	/home1/t/tst102
i5pool/home1/t/tst106	204800	56	204744	1%	/home1/t/tst106
i5pool/home1/t/tst11	204800	56	204744	1%	/home1/t/tst11
i5pool/home1/t/tst12	204800	57	204743	1%	/home1/t/tst12
i5pool/home1/t/tst17	204800	130	204670	1%	/home1/t/tst17
i5pool/home1/t/tst18	204800	59	204741	1%	/home1/t/tst18
i5pool/home1/t/tst2	204800	59	204741	1%	/home1/t/tst2
i5pool/home1/t/tst26	204800	59	204741	1%	/home1/t/tst26
i5pool/home1/v/vad224	204800	73985	130815	37%	/home1/v/vad224
i5pool/home1/v/vc317	204800	4692	200108	3%	/home1/v/vc317
i5pool/home1/v/vdt206	204800	25895	178905	13%	/home1/v/vdt206
i5pool/home1/v/vg14	204800	56	204744	1%	/home1/v/vg14
i5pool/home1/v/vh265	204800	3721	201079	2%	/home1/v/vh265
i5pool/home1/v/vj1	204800	56	204744	1%	/home1/v/vj1
i5pool/home1/v/vk18	204800	38584	166216	19%	/home1/v/vk18
i5pool/home1/v/vkd207	204800	6483	198317	4%	/home1/v/vkd207
i5pool/home1/v/vm545	307200	48261	258939	16%	/home1/v/vm545
i5pool/home1/v/vmd1	204800	56	204744	1%	/home1/v/vmd1
i5pool/home1/v/vms242	204800	55223	149577	27%	/home1/v/vms242
i5pool/home1/v/vnw200	204800	8777	196023	5%	/home1/v/vnw200
i5pool/home1/v/vpm214	204800	56	204744	1%	/home1/v/vpm214
i5pool/home1/v/vs31	204800	177	204623	1%	/home1/v/vs31
i5pool/home1/v/vs676	204800	56	204744	1%	/home1/v/vs676
i5pool/home1/v/vt1205	204800	56	204744	1%	/home1/v/vt1205
i5pool/home1/w/wc343	204800	59	204741	1%	/home1/w/wc343
i5pool/home1/w/wcw214	204800	56	204744	1%	/home1/w/wcw214
i5pool/home1/w/wew201	204800	849	203951	1%	/home1/w/wew201
i5pool/home1/w/wjy2	204800	46570	158230	23%	/home1/w/wjy2
i5pool/home1/w/wld215	204800	61	204739	1%	/home1/w/wld215
i5pool/home1/w/wnc200	204800	56	204744	1%	/home1/w/wnc200
i5pool/home1/w/wo6	204800	34382	170418	17%	/home1/w/wo6
i5pool/home1/w/wpb214	204800	72	204728	1%	/home1/w/wpb214
i5pool/home1/w/wtf3	204800	57	204743	1%	/home1/w/wtf3
i5pool/home1/x/xc213	204800	1507	203293	1%	/home1/x/xc213
i5pool/home1/x/xsc200	204800	56	204744	1%	/home1/x/xsc200
i5pool/home1/x/xwc202	204800	57370	147430	29%	/home1/x/xwc202
i5pool/home1/y/yc12	204800	59	204741	1%	/home1/y/yc12
i5pool/home1/y/ych248	204800	56	204744	1%	/home1/y/ych248
i5pool/home1/y/yci201	204800	60	204740	1%	/home1/y/yci201
i5pool/home1/y/yf342	204800	56	204744	1%	/home1/y/yf342
i5pool/home1/y/yl633	204800	56	204744	1%	/home1/y/yl633
i5pool/home1/y/yl1241	204800	10476	194324	6%	/home1/y/yl1241

i5pool/home1/y/yn247	204800	185	204615	1%	/home1/y/yn247
i5pool/home1/y/ync200	204800	1267	203533	1%	/home1/y/ync200
i5pool/home1/y/yr1	204800	110	204690	1%	/home1/y/yr1
i5pool/home1/y/yry203	204800	26121	178679	13%	/home1/y/yry203
i5pool/home1/y/yz256	204800	32367	172433	16%	/home1/y/yz256
i5pool/home1/y/yz437	204800	59	204741	1%	/home1/y/yz437
i5pool/home1/z/zk1	204800	56	204744	1%	/home1/z/zk1
i5pool/home1/z/zk269	204800	38637	166163	19%	/home1/z/zk269
i5pool/home1/z/zl226	204800	3696	201104	2%	/home1/z/zl226
i5pool/home1/z/zrd202	204800	56	204744	1%	/home1/z/zrd202
i5pool/home1/a/ad653	204800	141839	62961	70%	/home1/a/ad653
i5pool/home1/c/cwd3	102400	52	102348	1%	/home1/c/cwd3
i5pool/home1/j/jc198	102400	52	102348	1%	/home1/j/jc198
i5pool/home1/p/pjb249	102400	52	102348	1%	/home1/p/pjb249
i5pool/home1/m/mgp254	102400	52	102348	1%	/home1/m/mgp254
i5pool/home1/c/cdl267	102400	52	102348	1%	/home1/c/cdl267
i5pool/home1/m/mrs448	102400	52	102348	1%	/home1/m/mrs448
i5pool/home1/k/kz326	102400	52	102348	1%	/home1/k/kz326
i5pool/home1/j/jrf310	204800	61	204739	1%	/home1/j/jrf310
i5pool/home1/c/cer312	102400	52	102348	1%	/home1/c/cer312
i5pool/home1/e/emc390	102400	52	102348	1%	/home1/e/emc390
i5pool/home1/j/jsf291	102400	52	102348	1%	/home1/j/jsf291
i5pool/home1/a/ara269	102400	52	102348	1%	/home1/a/ara269
i5pool/home1/j/jfh258	102400	52	102348	1%	/home1/j/jfh258
i5pool/home1/d/djm403	204800	56	204744	1%	/home1/d/djm403
i5pool/home1/j/jkb278	102400	52	102348	1%	/home1/j/jkb278
i5pool/home1/k/ksb266	102400	52	102348	1%	/home1/k/ksb266
i5pool/home1/n/nrm265	102400	52	102348	1%	/home1/n/nrm265
i5pool/home1/b/bag270	204800	56	204744	1%	/home1/b/bag270
i5pool/home1/b/bdd219	102400	52	102348	1%	/home1/b/bdd219
i5pool/home1/n/nmh243	102400	52	102348	1%	/home1/n/nmh243
i5pool/home1/w/wrc226	102400	52	102348	1%	/home1/w/wrc226
i5pool/home1/k/ka1352	102400	52	102348	1%	/home1/k/ka1352
i5pool/home1/b/bmm278	102400	52	102348	1%	/home1/b/bmm278
i5pool/home1/h/hjr228	102400	52	102348	1%	/home1/h/hjr228
i5pool/home1/e/et660	102400	52	102348	1%	/home1/e/et660
i5pool/home1/k/kvc208	102400	52	102348	1%	/home1/k/kvc208
i5pool/home1/a/acw288	102400	52	102348	1%	/home1/a/acw288
i5pool/home1/t/tab304	102400	52	102348	1%	/home1/t/tab304
i5pool/home1/b/bcr230	102400	52	102348	1%	/home1/b/bcr230
i5pool/home1/j/jbr313	102400	52	102348	1%	/home1/j/jbr313
i5pool/home1/v/vhr205	102400	52	102348	1%	/home1/v/vhr205
i5pool/home1/d/dmg398	102400	52	102348	1%	/home1/d/dmg398
i5pool/home1/j/jav287	102400	52	102348	1%	/home1/j/jav287
i5pool/home1/a/ajm462	102400	52	102348	1%	/home1/a/ajm462
i5pool/home1/e/epl237	102400	52	102348	1%	/home1/e/epl237
i5pool/home1/r/rd889	102400	52	102348	1%	/home1/r/rd889
i5pool/home1/t/ttc230	102400	52	102348	1%	/home1/t/ttc230
i5pool/home1/t/tpm249	102400	52	102348	1%	/home1/t/tpm249
i5pool/home1/p/pay208	102400	52	102348	1%	/home1/p/pay208
i5pool/home1/m/mtz211	102400	52	102348	1%	/home1/m/mtz211
i5pool/home1/l/lge207	102400	52	102348	1%	/home1/l/lge207
i5pool/home1/k/ks1538	102400	52	102348	1%	/home1/k/ks1538
i5pool/home1/m/mpw253	102400	52	102348	1%	/home1/m/mpw253

i5pool/home1/j/jct252	102400	52	102348	1%	/home1/j/jct252
i5pool/home1/h/hac255	102400	52	102348	1%	/home1/h/hac255
i5pool/home1/l/lfa220	102400	52	102348	1%	/home1/l/lfa220
i5pool/home1/j/jf1437	102400	52	102348	1%	/home1/j/jf1437
i5pool/home1/j/jb1291	102400	52	102348	1%	/home1/j/jb1291
i5pool/home1/d/djl351	102400	52	102348	1%	/home1/d/djl351
i5pool/home1/y/yg373	102400	52	102348	1%	/home1/y/yg373
i5pool/home1/j/jcs476	102400	52	102348	1%	/home1/j/jcs476
i5pool/home1/d/dlk298	102400	52	102348	1%	/home1/d/dlk298
i5pool/home1/n/ncb247	102400	52	102348	1%	/home1/n/ncb247
i5pool/home1/a/as3931	102400	52	102348	1%	/home1/a/as3931
i5pool/home1/s/sp1302	102400	52	102348	1%	/home1/s/sp1302
i5pool/home1/a/acz223	102400	52	102348	1%	/home1/a/acz223
i5pool/home1/w/wc550	102400	52	102348	1%	/home1/w/wc550
i5pool/home1/j/jas800	102400	52	102348	1%	/home1/j/jas800
i5pool/home1/r/rms456	102400	52	102348	1%	/home1/r/rms456
i5pool/home1/c/cpl254	102400	52	102348	1%	/home1/c/cpl254
i5pool/home1/j/jt84	102400	52	102348	1%	/home1/j/jt84
i5pool/home1/a/acv226	102400	52	102348	1%	/home1/a/acv226
i5pool/home1/j/je666	102400	52	102348	1%	/home1/j/je666
i5pool/home1/g/gv357	102400	52	102348	1%	/home1/g/gv357
i5pool/home1/a/ajb479	102400	52	102348	1%	/home1/a/ajb479
i5pool/home1/a/aav228	102400	52	102348	1%	/home1/a/aav228
i5pool/home1/j/jmc662	102400	52	102348	1%	/home1/j/jmc662
i5pool/home1/w/wpg207	102400	52	102348	1%	/home1/w/wpg207
i5pool/home1/n/nm888	102400	52	102348	1%	/home1/n/nm888
i5pool/home1/m/mh960	204800	56	204744	1%	/home1/m/mh960
i5pool/home1/i/idk210	102400	52	102348	1%	/home1/i/idk210
i5pool/home1/j/jtb292	102400	52	102348	1%	/home1/j/jtb292
i5pool/home1/p/pjo247	102400	52	102348	1%	/home1/p/pjo247
i5pool/home1/d/dab447	102400	52	102348	1%	/home1/d/dab447
i5pool/home1/m/mrm421	102400	52	102348	1%	/home1/m/mrm421
i5pool/home1/v/vt394	102400	52	102348	1%	/home1/v/vt394
i5pool/home1/d/dam417	204800	56	204744	1%	/home1/d/dam417
i5pool/home1/m/mlg366	102400	52	102348	1%	/home1/m/mlg366
i5pool/home1/z/zk273	102400	52	102348	1%	/home1/z/zk273
i5pool/home1/n/nmt223	102400	52	102348	1%	/home1/n/nmt223
i5pool/home1/m/mcs378	102400	52	102348	1%	/home1/m/mcs378
i5pool/home1/m/mns268	102400	52	102348	1%	/home1/m/mns268
i5pool/home1/t/twf217	102400	52	102348	1%	/home1/t/twf217
i5pool/home1/x/xj223	102400	52	102348	1%	/home1/x/xj223
i5pool/home1/j/jl1402	102400	52	102348	1%	/home1/j/jl1402
i5pool/home1/a/an1233	102400	52	102348	1%	/home1/a/an1233
i5pool/home1/m/mc2695	102400	52	102348	1%	/home1/m/mc2695
i5pool/home1/d/djf284	102400	52	102348	1%	/home1/d/djf284
i5pool/home1/j/jem461	102400	52	102348	1%	/home1/j/jem461
i5pool/home1/z/zg265	204800	56	204744	1%	/home1/z/zg265
i5pool/home1/i/il450	102400	52	102348	1%	/home1/i/il450
i5pool/home1/b/bcw242	102400	52	102348	1%	/home1/b/bcw242
i5pool/home1/c/cz360	102400	52	102348	1%	/home1/c/cz360
i5pool/home1/b/bjl285	102400	52	102348	1%	/home1/b/bjl285
i5pool/home1/s/smr383	102400	52	102348	1%	/home1/s/smr383
i5pool/home1/e/ewr213	102400	52	102348	1%	/home1/e/ewr213
i5pool/home1/c/ca850	204800	12865	191935	7%	/home1/c/ca850

i5pool/home1/d/dlt258	204800	56	204744	1%	/home1/d/dlt258
i5pool/home1/c/cg995	204800	77976	126824	39%	/home1/c/cg995
i5pool/home1/a/ak1135	204800	56	204744	1%	/home1/a/ak1135
i5pool/home1/n/nys212	102400	52	102348	1%	/home1/n/nys212
i5pool/home1/a/ac1839	204800	56	204744	1%	/home1/a/ac1839
i5pool/home1/a/ajs518	102400	52	102348	1%	/home1/a/ajs518
i5pool/home1/g/gfs232	102400	52	102348	1%	/home1/g/gfs232
i5pool/home1/a/alk280	102400	52	102348	1%	/home1/a/alk280
i5pool/home1/c/cm332	204800	62024	142776	31%	/home1/c/cm332
i5pool/home1/z/zm280	102400	52	102348	1%	/home1/z/zm280
i5pool/home1/s/syp238	102400	52	102348	1%	/home1/s/syp238
i5pool/home1/m/myg209	102400	52	102348	1%	/home1/m/myg209
i5pool/home1/a/ars397	204800	56	204744	1%	/home1/a/ars397
i5pool/home1/s/sr1641	102400	52	102348	1%	/home1/s/sr1641
i5pool/home1/c/ccp257	204800	56	204744	1%	/home1/c/ccp257
i5pool/home1/d/du235	102400	52	102348	1%	/home1/d/du235
i5pool/home1/s/sys245	102400	52	102348	1%	/home1/s/sys245
i5pool/home1/d/deg294	102400	52	102348	1%	/home1/d/deg294
i5pool/home1/j/jl1924	102400	52	102348	1%	/home1/j/jl1924
i5pool/home1/s/ssf243	102400	52	102348	1%	/home1/s/ssf243
i5pool/home1/c/cm508	102400	52	102348	1%	/home1/c/cm508
i5pool/home1/r/rx208	102400	52	102348	1%	/home1/r/rx208
i5pool/home1/s/sz439	102400	52	102348	1%	/home1/s/sz439
i5pool/home1/a/ask304	102400	52	102348	1%	/home1/a/ask304
i5pool/home1/l/ljk268	102400	52	102348	1%	/home1/l/ljk268
i5pool/home1/r/rva212	204800	5559	199241	3%	/home1/r/rva212
i5pool/home1/m/mam824	102400	52	102348	1%	/home1/m/mam824
i5pool/home1/n/ns1087	102400	52	102348	1%	/home1/n/ns1087
i5pool/home1/w/wcw212	102400	52	102348	1%	/home1/w/wcw212
i5pool/home1/k/kkk236	102400	52	102348	1%	/home1/k/kkk236
i5pool/home1/b/bd579	102400	52	102348	1%	/home1/b/bd579
i5pool/home1/s/sh1300	102400	52	102348	1%	/home1/s/sh1300
i5pool/home1/k/k1819	102400	52	102348	1%	/home1/k/k1819
i5pool/home1/k/kas513	102400	52	102348	1%	/home1/k/kas513
i5pool/home1/k/kmd322	102400	52	102348	1%	/home1/k/kmd322
i5pool/home1/y/yp339	102400	52	102348	1%	/home1/y/yp339
i5pool/home1/b/ba559	102400	52	102348	1%	/home1/b/ba559
i5pool/home1/h/hs1135	102400	52	102348	1%	/home1/h/hs1135
i5pool/home1/j/jp1717	102400	52	102348	1%	/home1/j/jp1717
i5pool/home1/r/rc1421	204800	56	204744	1%	/home1/r/rc1421
i5pool/home1/m/mc2996	102400	52	102348	1%	/home1/m/mc2996
i5pool/home1/h/hw488	102400	52	102348	1%	/home1/h/hw488
i5pool/home1/b/bmh242	102400	52	102348	1%	/home1/b/bmh242
i5pool/home1/c/cl1123	102400	52	102348	1%	/home1/c/cl1123
i5pool/home1/k/kav1	102400	52	102348	1%	/home1/k/kav1
i5pool/home1/a/aom223	102400	52	102348	1%	/home1/a/aom223
i5pool/home1/j/jmg635	102400	52	102348	1%	/home1/j/jmg635
i5pool/home1/a/af1026	204800	96363	108437	48%	/home1/a/af1026
i5pool/home1/b/byw209	102400	52	102348	1%	/home1/b/byw209
i5pool/home1/m/mjv250	204800	68282	136518	34%	/home1/m/mjv250
i5pool/home1/k/kmp367	102400	52	102348	1%	/home1/k/kmp367
i5pool/home1/a/apm282	102400	52	102348	1%	/home1/a/apm282
i5pool/home1/m/mm4130	102400	52	102348	1%	/home1/m/mm4130
i5pool/home1/j/jp1311	102400	52	102348	1%	/home1/j/jp1311

i5pool/home1/m/mps318	102400	52	102348	1%	/home1/m/mps318
i5pool/home1/r/rfw227	102400	52	102348	1%	/home1/r/rfw227
i5pool/home1/m/mws262	102400	52	102348	1%	/home1/m/mws262
i5pool/home1/m/may240	204800	83365	121435	41%	/home1/m/may240
i5pool/home1/b/bjl278	102400	52	102348	1%	/home1/b/bjl278
i5pool/home1/a/ast243	204800	27698	177102	14%	/home1/a/ast243
i5pool/home1/j/jjh360	102400	52	102348	1%	/home1/j/jjh360
i5pool/home1/y/yc527	102400	52	102348	1%	/home1/y/yc527
i5pool/home1/j/jcs414	102400	52	102348	1%	/home1/j/jcs414
i5pool/home1/s/sdf259	102400	52	102348	1%	/home1/s/sdf259
i5pool/home1/a/aas424	102400	50	102350	1%	/home1/a/aas424
i5pool/home1/j/jel385	102400	52	102348	1%	/home1/j/jel385
i5pool/home1/c/cmo267	102400	52	102348	1%	/home1/c/cmo267
i5pool/home1/v/veg216	102400	52	102348	1%	/home1/v/veg216
i5pool/home1/r/rs1959	102400	52	102348	1%	/home1/r/rs1959
i5pool/home1/p/pbp227	102400	52	102348	1%	/home1/p/pbp227
i5pool/home1/l/lh1218	204800	56	204744	1%	/home1/l/lh1218
i5pool/home1/j/jh1939	102400	52	102348	1%	/home1/j/jh1939
i5pool/home1/j/jec394	102400	52	102348	1%	/home1/j/jec394
i5pool/home1/m/mba238	204800	56	204744	1%	/home1/m/mba238
i5pool/home1/d/dsm305	102400	52	102348	1%	/home1/d/dsm305
i5pool/home1/w/wjc269	102400	52	102348	1%	/home1/w/wjc269
i5pool/home1/r/rmd293	102400	52	102348	1%	/home1/r/rmd293
i5pool/home1/s/sa1100	204800	27796	177004	14%	/home1/s/sa1100
i5pool/home1/d/djc366	102400	52	102348	1%	/home1/d/djc366
i5pool/home1/f/fjs246	204800	21762	183038	11%	/home1/f/fjs246
i5pool/home1/a/aeb337	102400	50	102350	1%	/home1/a/aeb337
i5pool/home1/a/acg333	102400	52	102348	1%	/home1/a/acg333
i5pool/home1/n/nsm257	102400	52	102348	1%	/home1/n/nsm257
i5pool/home1/m/mdk298	102400	52	102348	1%	/home1/m/mdk298
i5pool/home1/a/arm399	102400	52	102348	1%	/home1/a/arm399
i5pool/home1/m/mrm396	102400	52	102348	1%	/home1/m/mrm396
i5pool/home1/j/jmw447	102400	52	102348	1%	/home1/j/jmw447
i5pool/home1/a/asm327	204800	56	204744	1%	/home1/a/asm327
i5pool/home1/d/drs343	102400	52	102348	1%	/home1/d/drs343
i5pool/home1/v/vjp215	204800	56	204744	1%	/home1/v/vjp215
i5pool/home1/a/aj714	102400	52	102348	1%	/home1/a/aj714
i5pool/home1/d/dr975	204800	56	204744	1%	/home1/d/dr975
i5pool/home1/e/erp205	102400	52	102348	1%	/home1/e/erp205
i5pool/home1/f/fb538	102400	52	102348	1%	/home1/f/fb538
i5pool/home1/j/jc2710	102400	52	102348	1%	/home1/j/jc2710
i5pool/home1/d/dw893	102400	52	102348	1%	/home1/d/dw893
i5pool/home1/n/nl587	204800	52277	152523	26%	/home1/n/nl587
i5pool/home1/s/stv207	102400	52	102348	1%	/home1/s/stv207
i5pool/home1/c/cl1193	102400	52	102348	1%	/home1/c/cl1193
i5pool/home1/e/em1456	102400	52	102348	1%	/home1/e/em1456
i5pool/home1/a/ahw243	102400	52	102348	1%	/home1/a/ahw243
i5pool/home1/r/rmt283	102400	52	102348	1%	/home1/r/rmt283
i5pool/home1/k/k1926	102400	52	102348	1%	/home1/k/k1926
i5pool/home1/h/hj364	102400	52	102348	1%	/home1/h/hj364
i5pool/home1/j/jpb267	102400	52	102348	1%	/home1/j/jpb267
i5pool/home1/g/gms325	102400	52	102348	1%	/home1/g/gms325
i5pool/home1/j/jsb448	102400	52	102348	1%	/home1/j/jsb448
i5pool/home1/j/jd1506	102400	52	102348	1%	/home1/j/jd1506

i5pool/home1/s/scw271	102400	52	102348	1%	/home1/s/scw271
i5pool/home1/b/br1231	204800	54682	150118	27%	/home1/b/br1231
i5pool/home1/n/nmc259	102400	52	102348	1%	/home1/n/nmc259
i5pool/home1/b/bag274	102400	52	102348	1%	/home1/b/bag274
i5pool/home1/m/mmc386	204800	56	204744	1%	/home1/m/mmc386
i5pool/home1/l/lef275	102400	52	102348	1%	/home1/l/lef275
i5pool/home1/s/sjk361	102400	52	102348	1%	/home1/s/sjk361
i5pool/home1/d/dna213	102400	52	102348	1%	/home1/d/dna213
i5pool/home1/i/idr209	102400	52	102348	1%	/home1/i/idr209
i5pool/home1/c/cmb433	102400	52	102348	1%	/home1/c/cmb433
i5pool/home1/m/mhl277	102400	52	102348	1%	/home1/m/mhl277
i5pool/home1/v/vz254	102400	52	102348	1%	/home1/v/vz254
i5pool/home1/t/tmo219	307200	241705	65495	79%	/home1/t/tmo219
i5pool/home1/j/jac607	102400	52	102348	1%	/home1/j/jac607
i5pool/home1/r/rmo231	102400	52	102348	1%	/home1/r/rmo231
i5pool/home1/b/bl1238	102400	52	102348	1%	/home1/b/bl1238
i5pool/home1/j/jds434	204800	56	204744	1%	/home1/j/jds434
i5pool/home1/j/jt1116	102400	52	102348	1%	/home1/j/jt1116
i5pool/home1/c/cec359	102400	52	102348	1%	/home1/c/cec359
i5pool/home1/k/ks1278	102400	52	102348	1%	/home1/k/ks1278
i5pool/home1/a/ah1255	102400	52	102348	1%	/home1/a/ah1255
i5pool/home1/k/klt249	102400	52	102348	1%	/home1/k/klt249
i5pool/home1/a/aja312	102400	52	102348	1%	/home1/a/aja312
i5pool/home1/a/amp438	102400	52	102348	1%	/home1/a/amp438
i5pool/home1/p/pbw218	102400	52	102348	1%	/home1/p/pbw218
i5pool/home1/g/gpb226	102400	52	102348	1%	/home1/g/gpb226
i5pool/home1/a/ana247	102400	52	102348	1%	/home1/a/ana247
i5pool/home1/t/tej214	102400	52	102348	1%	/home1/t/tej214
i5pool/home1/a/ac1859	102400	52	102348	1%	/home1/a/ac1859
i5pool/home1/a/ar1301	204800	288	204512	1%	/home1/a/ar1301
i5pool/home1/r/rem356	102400	52	102348	1%	/home1/r/rem356
i5pool/home1/e/eh798	204800	56	204744	1%	/home1/e/eh798
i5pool/home1/s/smd352	102400	52	102348	1%	/home1/s/smd352
i5pool/home1/s/shk338	102400	52	102348	1%	/home1/s/shk338
i5pool/home1/f/fmh221	102400	52	102348	1%	/home1/f/fmh221
i5pool/home1/m/mtd292	102400	52	102348	1%	/home1/m/mtd292
i5pool/home1/q/qcm201	102400	52	102348	1%	/home1/q/qcm201
i5pool/home1/j/jea275	102400	52	102348	1%	/home1/j/jea275
i5pool/home1/s/sal356	102400	52	102348	1%	/home1/s/sal356
i5pool/home1/e/emc434	102400	52	102348	1%	/home1/e/emc434
i5pool/home1/s/swr225	102400	52	102348	1%	/home1/s/swr225
i5pool/home1/c/cch302	102400	52	102348	1%	/home1/c/cch302
i5pool/home1/b/bhm229	102400	52	102348	1%	/home1/b/bhm229
i5pool/home1/s/ses464	102400	52	102348	1%	/home1/s/ses464
i5pool/home1/a/ac1902	102400	52	102348	1%	/home1/a/ac1902
i5pool/home1/j/jds460	102400	52	102348	1%	/home1/j/jds460
i5pool/home1/m/mc2711	204800	56	204744	1%	/home1/m/mc2711
i5pool/home1/r/rw808	102400	52	102348	1%	/home1/r/rw808
i5pool/home1/e/ewj203	102400	52	102348	1%	/home1/e/ewj203
i5pool/home1/p/plr236	102400	52	102348	1%	/home1/p/plr236
i5pool/home1/b/btm246	102400	52	102348	1%	/home1/b/btm246
i5pool/home1/d/dmc441	102400	52	102348	1%	/home1/d/dmc441
i5pool/home1/m/maw410	102400	52	102348	1%	/home1/m/maw410
i5pool/home1/e/ers321	102400	52	102348	1%	/home1/e/ers321

i5pool/home1/k/kcv211	102400	52	102348	1%	/home1/k/kcv211
i5pool/home1/y/yhc275	102400	52	102348	1%	/home1/y/yhc275
i5pool/home1/m/mbc295	102400	52	102348	1%	/home1/m/mbc295
i5pool/home1/c/cwb249	102400	52	102348	1%	/home1/c/cwb249
i5pool/home1/s/srg323	102400	52	102348	1%	/home1/s/srg323
i5pool/home1/b/bos207	204800	22484	182316	11%	/home1/b/bos207
i5pool/home1/g/gns218	204800	83964	120836	41%	/home1/g/gns218
i5pool/home1/s/scr274	102400	52	102348	1%	/home1/s/scr274
i5pool/home1/n/ns1005	102400	52	102348	1%	/home1/n/ns1005
i5pool/home1/h/hnm206	102400	52	102348	1%	/home1/h/hnm206
i5pool/home1/m/mpn230	102400	52	102348	1%	/home1/m/mpn230
i5pool/home1/e/eg979	102400	52	102348	1%	/home1/e/eg979
i5pool/home1/w/wpw208	204800	37596	167204	19%	/home1/w/wpw208
i5pool/home1/r/ra870	204800	863	203937	1%	/home1/r/ra870
i5pool/home1/j/ja1212	102400	52	102348	1%	/home1/j/ja1212
i5pool/home1/b/bhr209	102400	52	102348	1%	/home1/b/bhr209
i5pool/home1/c/ca820	204800	59183	145617	29%	/home1/c/ca820
i5pool/home1/a/asm389	102400	52	102348	1%	/home1/a/asm389
i5pool/home1/m/mah484	204800	56	204744	1%	/home1/m/mah484
i5pool/home1/a/ajj241	102400	52	102348	1%	/home1/a/ajj241
i5pool/home1/p/psj214	102400	52	102348	1%	/home1/p/psj214
i5pool/home1/w/wma216	102400	52	102348	1%	/home1/w/wma216
i5pool/home1/b/bpd223	102400	52	102348	1%	/home1/b/bpd223
i5pool/home1/c/cot206	102400	52	102348	1%	/home1/c/cot206
i5pool/home1/g/gbw227	204800	56	204744	1%	/home1/g/gbw227
i5pool/home1/r/rer264	204800	56	204744	1%	/home1/r/rer264
i5pool/home1/j/jlb493	102400	52	102348	1%	/home1/j/jlb493
i5pool/home1/n/nrs249	102400	52	102348	1%	/home1/n/nrs249
i5pool/home1/a/agp234	102400	52	102348	1%	/home1/a/agp234
i5pool/home1/m/mbk274	102400	52	102348	1%	/home1/m/mbk274
i5pool/home1/b/bd581	102400	52	102348	1%	/home1/b/bd581
i5pool/home1/j/jho217	102400	52	102348	1%	/home1/j/jho217
i5pool/home1/j/jc2955	204800	56	204744	1%	/home1/j/jc2955
i5pool/home1/a/aad309	102400	52	102348	1%	/home1/a/aad309
i5pool/home1/a/af1047	102400	52	102348	1%	/home1/a/af1047
i5pool/home1/j/jc2628	204800	56	204744	1%	/home1/j/jc2628
i5pool/home1/j/jag485	102400	52	102348	1%	/home1/j/jag485
i5pool/home1/s/sbh297	102400	52	102348	1%	/home1/s/sbh297
i5pool/home1/e/ems409	102400	52	102348	1%	/home1/e/ems409
i5pool/home1/a/ajv238	102400	52	102348	1%	/home1/a/ajv238
i5pool/home1/r/rtp232	102400	52	102348	1%	/home1/r/rtp232
i5pool/home1/s/smt308	102400	52	102348	1%	/home1/s/smt308
i5pool/home1/k/kmv236	102400	52	102348	1%	/home1/k/kmv236
i5pool/home1/a/ara252	102400	52	102348	1%	/home1/a/ara252
i5pool/home1/k/kmk400	102400	52	102348	1%	/home1/k/kmk400
i5pool/home1/m/mm294	102400	52	102348	1%	/home1/m/mm294
i5pool/home1/a/am2833	102400	52	102348	1%	/home1/a/am2833
i5pool/home1/m/mrp325	102400	52	102348	1%	/home1/m/mrp325
i5pool/home1/a/aln249	102400	52	102348	1%	/home1/a/aln249
i5pool/home1/j/jlr426	102400	52	102348	1%	/home1/j/jlr426
i5pool/home1/j/jcs467	102400	52	102348	1%	/home1/j/jcs467
i5pool/home1/s/sdr287	102400	52	102348	1%	/home1/s/sdr287
i5pool/home1/m/mmr345	102400	52	102348	1%	/home1/m/mmr345
i5pool/home1/k/kzr202	102400	52	102348	1%	/home1/k/kzr202

i5pool/home1/s/sts290	102400	52	102348	1%	/home1/s/sts290
i5pool/home1/s/sbm304	102400	52	102348	1%	/home1/s/sbm304
i5pool/home1/a/aym211	102400	52	102348	1%	/home1/a/aym211
i5pool/home1/c/cn1223	102400	52	102348	1%	/home1/c/cn1223
i5pool/home1/s/sma306	102400	52	102348	1%	/home1/s/sma306
i5pool/home1/b/bkm224	204800	56	204744	1%	/home1/b/bkm224
i5pool/home1/j/jlm501	102400	52	102348	1%	/home1/j/jlm501
i5pool/home1/a/amd448	102400	52	102348	1%	/home1/a/amd448
i5pool/home1/e/epc251	102400	52	102348	1%	/home1/e/epc251
i5pool/home1/n/nmu204	102400	52	102348	1%	/home1/n/nmu204
i5pool/home1/m/mec433	102400	52	102348	1%	/home1/m/mec433
i5pool/home1/a/aw984	307200	255500	51700	84%	/home1/a/aw984
i5pool/home1/j/jwj240	102400	52	102348	1%	/home1/j/jwj240
i5pool/home1/n/npt212	102400	52	102348	1%	/home1/n/npt212
i5pool/home1/j/jaa398	102400	52	102348	1%	/home1/j/jaa398
i5pool/home1/g/gs1232	102400	52	102348	1%	/home1/g/gs1232
i5pool/home1/n/nmd246	204800	56	204744	1%	/home1/n/nmd246
i5pool/home1/e/eec266	102400	52	102348	1%	/home1/e/eec266
i5pool/home1/s/src296	102400	52	102348	1%	/home1/s/src296
i5pool/home1/s/set270	204800	56	204744	1%	/home1/s/set270
i5pool/home1/r/r11173	102400	52	102348	1%	/home1/r/r11173
i5pool/home1/s/seg345	102400	52	102348	1%	/home1/s/seg345
i5pool/home1/a/ash302	102400	52	102348	1%	/home1/a/ash302
i5pool/home1/j/jnc248	102400	52	102348	1%	/home1/j/jnc248
i5pool/home1/b/baj232	102400	52	102348	1%	/home1/b/baj232
i5pool/home1/d/dck244	102400	52	102348	1%	/home1/d/dck244
i5pool/home1/b/be377	102400	52	102348	1%	/home1/b/be377
i5pool/home1/j/jrc400	102400	52	102348	1%	/home1/j/jrc400
i5pool/home1/m/ml1344	102400	52	102348	1%	/home1/m/ml1344
i5pool/home1/a/amd418	102400	52	102348	1%	/home1/a/amd418
i5pool/home1/t/tmb283	102400	52	102348	1%	/home1/t/tmb283
i5pool/home1/m/mpw256	102400	52	102348	1%	/home1/m/mpw256
i5pool/home1/r/ram435	102400	52	102348	1%	/home1/r/ram435
i5pool/home1/t/tt665	102400	52	102348	1%	/home1/t/tt665
i5pool/home1/x/xd223	102400	52	102348	1%	/home1/x/xd223
i5pool/home1/a/ao631	204800	60	204740	1%	/home1/a/ao631
i5pool/home1/a/ar1677	102400	52	102348	1%	/home1/a/ar1677
i5pool/home1/m/mk2648	204800	87246	117554	43%	/home1/m/mk2648
i5pool/home1/s/sh1293	102400	52	102348	1%	/home1/s/sh1293
i5pool/home1/a/ar1581	102400	52	102348	1%	/home1/a/ar1581
i5pool/home1/r/rmj258	102400	52	102348	1%	/home1/r/rmj258
i5pool/home1/y/yjc261	102400	52	102348	1%	/home1/y/yjc261
i5pool/home1/s/sjh321	102400	52	102348	1%	/home1/s/sjh321
i5pool/home1/n/nhp216	102400	52	102348	1%	/home1/n/nhp216
i5pool/home1/r/ra979	102400	52	102348	1%	/home1/r/ra979
i5pool/home1/s/shm258	102400	52	102348	1%	/home1/s/shm258
i5pool/home1/j/jy1319	102400	52	102348	1%	/home1/j/jy1319
i5pool/home1/b/bjk279	102400	52	102348	1%	/home1/b/bjk279
i5pool/home1/j/jyk304	102400	52	102348	1%	/home1/j/jyk304
i5pool/home1/j/jy542	102400	52	102348	1%	/home1/j/jy542
i5pool/home1/j/jch379	102400	52	102348	1%	/home1/j/jch379
i5pool/home1/j/jp1719	102400	52	102348	1%	/home1/j/jp1719
i5pool/home1/a/ajk377	102400	52	102348	1%	/home1/a/ajk377
i5pool/home1/e/ec1259	102400	52	102348	1%	/home1/e/ec1259



i5pool/home1/w/ww435	102400	52	102348	1%	/home1/w/ww435
i5pool/home1/j/jh1386	102400	52	102348	1%	/home1/j/jh1386
i5pool/home1/h/hk869	102400	52	102348	1%	/home1/h/hk869
i5pool/home1/j/js4292	102400	52	102348	1%	/home1/j/js4292
i5pool/home1/a/ag842	102400	52	102348	1%	/home1/a/ag842
i5pool/home1/k/kst232	102400	52	102348	1%	/home1/k/kst232
i5pool/home1/g/gkm210	102400	52	102348	1%	/home1/g/gkm210
i5pool/home1/k/kn529	102400	52	102348	1%	/home1/k/kn529
i5pool/home1/h/hjy231	102400	52	102348	1%	/home1/h/hjy231
i5pool/home1/a/aea270	204800	56	204744	1%	/home1/a/aea270
i5pool/home1/j/jj1390	102400	52	102348	1%	/home1/j/jj1390
i5pool/home1/c/cab484	102400	52	102348	1%	/home1/c/cab484
i5pool/home1/i/icp205	204800	5038	199762	3%	/home1/i/icp205
i5pool/home1/l/lmc415	102400	52	102348	1%	/home1/l/lmc415
i5pool/home1/k/kab414	204800	56	204744	1%	/home1/k/kab414
i5pool/home1/m/mv489	204800	8216	196584	5%	/home1/m/mv489
i5pool/home1/a/ajb481	102400	52	102348	1%	/home1/a/ajb481
i5pool/home1/n/nnm221	102400	52	102348	1%	/home1/n/nnm221
i5pool/home1/t/tr633	102400	52	102348	1%	/home1/t/tr633
i5pool/home1/a/abn212	102400	52	102348	1%	/home1/a/abn212
/usr/local	69901096	4665044	64537042	7%	/usr/local
i5pool/home1/j/jls674	102400	52	102348	1%	/home1/j/jls674
i5pool/home1/h/hhh240	102400	52	102348	1%	/home1/h/hhh240
i5pool/home1/o/osh206	204800	7576	197224	4%	/home1/o/osh206
i5pool/home1/o/omb210	102400	52	102348	1%	/home1/o/omb210
i5pool/home1/m/mah482	204800	56	204744	1%	/home1/m/mah482
i5pool/home1/p/pmc305	102400	52	102348	1%	/home1/p/pmc305
i5pool/home1/m/mjm605	102400	52	102348	1%	/home1/m/mjm605
i5pool/home1/g/gl634	102400	52	102348	1%	/home1/g/gl634
i5pool/home1/j/jld357	102400	52	102348	1%	/home1/j/jld357
i5pool/home1/d/dmb443	102400	52	102348	1%	/home1/d/dmb443
i5pool/home1/p/pp872	102400	52	102348	1%	/home1/p/pp872
i5pool/home1/s/sab453	204800	28085	176715	14%	/home1/s/sab453
i5pool/home1/t/tes271	204800	56	204744	1%	/home1/t/tes271
i5pool/home1/v/vlm238	102400	52	102348	1%	/home1/v/vlm238
i5pool/home1/l/lje222	102400	52	102348	1%	/home1/l/lje222
i5pool/home1/i/im549	204800	126178	78622	62%	/home1/i/im549
i5pool/home1/a/ats279	102400	52	102348	1%	/home1/a/ats279
i5pool/home1/n/nmc295	102400	52	102348	1%	/home1/n/nmc295
i5pool/home1/j/jec372	102400	52	102348	1%	/home1/j/jec372
i5pool/home1/k/kcr239	102400	52	102348	1%	/home1/k/kcr239
i5pool/home1/a/aep282	102400	52	102348	1%	/home1/a/aep282
i5pool/home1/v/vi245	102400	52	102348	1%	/home1/v/vi245
i5pool/home1/j/jdw329	102400	52	102348	1%	/home1/j/jdw329
i5pool/home1/p/pmm315	102400	52	102348	1%	/home1/p/pmm315
i5pool/home1/l/lw827	102400	52	102348	1%	/home1/l/lw827
i5pool/home1/j/jey231	102400	52	102348	1%	/home1/j/jey231
i5pool/home1/t/tr599	102400	52	102348	1%	/home1/t/tr599
i5pool/home1/t/tn455	102400	52	102348	1%	/home1/t/tn455
i5pool/home1/k/kjm326	102400	52	102348	1%	/home1/k/kjm326
i5pool/home1/q/qh221	102400	52	102348	1%	/home1/q/qh221
i5pool/home1/b/bam344	102400	52	102348	1%	/home1/b/bam344
i5pool/home1/t/ts1241	102400	52	102348	1%	/home1/t/ts1241
i5pool/home1/j/jhh309	204800	56	204744	1%	/home1/j/jhh309

i5pool/home1/e/ed673	102400	52	102348	1%	/home1/e/ed673
i5pool/home1/j/jz545	102400	52	102348	1%	/home1/j/jz545
i5pool/home1/s/sg1779	102400	52	102348	1%	/home1/s/sg1779
i5pool/home1/t/tc761	204800	82927	121873	41%	/home1/t/tc761
i5pool/home1/j/jmc745	102400	52	102348	1%	/home1/j/jmc745
i5pool/home1/m/mf1428	102400	52	102348	1%	/home1/m/mf1428
i5pool/home1/s/seh320	102400	52	102348	1%	/home1/s/seh320
i5pool/home1/a/adb330	102400	52	102348	1%	/home1/a/adb330
i5pool/home1/b/bal273	204800	13215	191585	7%	/home1/b/bal273
i5pool/home1/b/bar300	102400	52	102348	1%	/home1/b/bar300
i5pool/home1/p/ppk212	102400	52	102348	1%	/home1/p/ppk212
i5pool/home1/k/krs315	102400	52	102348	1%	/home1/k/krs315
i5pool/home1/d/dv461	204800	56	204744	1%	/home1/d/dv461
i5pool/home1/s/ss4063	102400	52	102348	1%	/home1/s/ss4063
i5pool/home1/m/mh986	102400	52	102348	1%	/home1/m/mh986
i5pool/home1/r/rms441	102400	52	102348	1%	/home1/r/rms441
i5pool/home1/g/gmc269	102400	52	102348	1%	/home1/g/gmc269
i5pool/home1/k/km1497	102400	52	102348	1%	/home1/k/km1497
i5pool/home1/t/tjb284	102400	52	102348	1%	/home1/t/tjb284
i5pool/home1/j/jhp324	102400	52	102348	1%	/home1/j/jhp324
i5pool/home1/e/em1358	102400	52	102348	1%	/home1/e/em1358
i5pool/home1/k/ks1829	102400	52	102348	1%	/home1/k/ks1829
i5pool/home1/d/did212	102400	52	102348	1%	/home1/d/did212
i5pool/home1/b/bck226	102400	52	102348	1%	/home1/b/bck226
i5pool/home1/n/nrd216	102400	52	102348	1%	/home1/n/nrd216
i5pool/home1/a/aeg312	204800	8570	196230	5%	/home1/a/aeg312
i5pool/home1/m/mag570	102400	52	102348	1%	/home1/m/mag570
i5pool/home1/j/jm1447	102400	52	102348	1%	/home1/j/jm1447
i5pool/home1/c/cz402	102400	52	102348	1%	/home1/c/cz402
i5pool/home1/c/cem338	102400	52	102348	1%	/home1/c/cem338
i5pool/home1/r/rc1266	102400	52	102348	1%	/home1/r/rc1266
i5pool/home1/y/yw379	102400	52	102348	1%	/home1/y/yw379
i5pool/home1/f/fm663	102400	52	102348	1%	/home1/f/fm663
i5pool/home1/j/jp1506	102400	52	102348	1%	/home1/j/jp1506
i5pool/home1/j/jd1363	102400	52	102348	1%	/home1/j/jd1363
i5pool/home1/j/jdd295	102400	52	102348	1%	/home1/j/jdd295
i5pool/home1/s/sj638	102400	52	102348	1%	/home1/s/sj638
i5pool/home1/c/cmm496	102400	52	102348	1%	/home1/c/cmm496
i5pool/home1/r/rla252	102400	52	102348	1%	/home1/r/rla252
i5pool/home1/h/hjs256	102400	52	102348	1%	/home1/h/hjs256
i5pool/home1/s/scs348	102400	52	102348	1%	/home1/s/scs348
i5pool/home1/g/gv358	204800	35053	169747	18%	/home1/g/gv358
i5pool/home1/s/sg1657	102400	52	102348	1%	/home1/s/sg1657
i5pool/home1/k/khh242	204800	21408	183392	11%	/home1/k/khh242
i5pool/home1/m/ml2502	102400	52	102348	1%	/home1/m/ml2502
i5pool/home1/s/shh270	204800	14720	190080	8%	/home1/s/shh270
i5pool/home1/s/srs397	204800	56	204744	1%	/home1/s/srs397
i5pool/home1/a/ahy210	307200	278868	28332	91%	/home1/a/ahy210
i5pool/home1/a/amc542	102400	52	102348	1%	/home1/a/amc542
i5pool/home1/e/erc267	102400	52	102348	1%	/home1/e/erc267
i5pool/home1/s/sz519	102400	52	102348	1%	/home1/s/sz519
i5pool/home1/k/kc1001	102400	52	102348	1%	/home1/k/kc1001
i5pool/home1/r/rr1263	204800	56	204744	1%	/home1/r/rr1263
i5pool/home1/j/jmf387	102400	52	102348	1%	/home1/j/jmf387

i5pool/home1/a/ams808	102400	52	102348	1%	/home1/a/ams808
i5pool/home1/l/lh937	102400	52	102348	1%	/home1/l/lh937
i5pool/home1/n/njh237	102400	50	102350	1%	/home1/n/njh237
i5pool/home1/d/dp951	102400	52	102348	1%	/home1/d/dp951
i5pool/home1/r/ras525	102400	52	102348	1%	/home1/r/ras525
i5pool/home1/m/mg2762	102400	52	102348	1%	/home1/m/mg2762
i5pool/home1/a/arg302	102400	50	102350	1%	/home1/a/arg302
i5pool/home1/n/ndc229	102400	52	102348	1%	/home1/n/ndc229
i5pool/home1/l/lnp218	102400	50	102350	1%	/home1/l/lnp218
i5pool/home1/d/drh269	102400	52	102348	1%	/home1/d/drh269
i5pool/home1/d/dc1401	102400	50	102350	1%	/home1/d/dc1401
i5pool/home1/m/man289	102400	50	102350	1%	/home1/m/man289
i5pool/home1/j/jc3181	102400	50	102350	1%	/home1/j/jc3181
i5pool/home1/m/mb2656	204800	56	204744	1%	/home1/m/mb2656
i5pool/home1/k/kmg319	102400	50	102350	1%	/home1/k/kmg319
i5pool/home1/h/hjm237	102400	50	102350	1%	/home1/h/hjm237
i5pool/home1/s/shc329	102400	52	102348	1%	/home1/s/shc329
i5pool/home1/m/ml1590	204800	20660	184140	11%	/home1/m/ml1590

```

10$ /usr/xpg4/bin/df -P | tail +2 | awk '{print $5, ($3 / $2) * 100}'
11$ /usr/xpg4/bin/df -P | tail +2 | awk '{print $5, 100 * $3 / $2}' simpler way to do same thing
25% 24.3307
25% 24.3307
46% 44.6638
46% 44.6638
46% 44.6638
46% 44.6638
46% 44.6638
46% 44.6638
46% 44.6638
46% 44.6638
0
0
0
0
1% 0.00174802
46% 44.6638
46% 44.6638
0
1% 0.439627
1% 0.000211884
46% 40.4462
1% 0.000294918
1% 0.000130362
1% 0.000151733
1% 0.000151733
1% 0.000120389
1% 4.9153e-05
1% 8.47711e-05
1% 7.90722e-05
1% 5.62766e-05
1% 0.00031059
1% 0.000126088
1% 8.47711e-05
1% 0.000257875
1% 0.00011469
1% 5.05777e-05
1% 9.04699e-05
1% 4.20293e-05
1% 0.000147459
1% 0.000209434
1% 0.000111841
1% 3.49057e-05
1% 7.90722e-05
1% 6.90991e-05
1% 4.48788e-05
1% 7.05238e-05
1% 4.77282e-05
1% 0.0288086
1% 0.598145
1% 0.0273438
1% 0.0273438
1% 0.027832
42% 41.8765

```

6% 5.26367  
1% 0.0273438  
1% 0.0273438  
2% 1.99854  
16% 15.3226  
2% 1.99951  
1% 0.0273438  
1% 0.0273438  
51% 50.1567  
11% 10.8047  
21% 20.5498  
25% 24.3486  
2% 1.71338  
1% 0.0317383  
1% 0.0273438  
1% 0.0273438  
22% 21.7905  
6% 5.15283  
1% 0.0273438  
1% 0.0366211  
1% 0.0273438  
1% 0.0317383  
1% 0.23584  
1% 0.0273438  
2% 1.26758  
1% 0.0273438  
1% 0.208008  
13% 12.1406  
1% 0.0273438  
1% 0.0273438  
38% 37.9961  
66% 65.4896  
46% 45.5708  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
50% 49.3784  
13% 12.5415  
1% 0.0273438  
1% 0.0273438  
4% 3.13965  
18% 17.0928  
1% 0.0273438  
50% 49.7905  
1% 0.604492  
1% 0.0273438  
13% 12.3179  
1% 0.0273438  
20% 19.5493  
12% 11.5493  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438

3% 2.66895  
1% 0.0273438  
1% 0.0273438  
12% 11.6152  
1% 0.0273438  
1% 0.0273438  
17% 16.772  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
37% 36.3081  
31% 30.0938  
13% 12.6904  
1% 0.0273438  
1% 0.0273438  
1% 0.027832  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
36% 35.5322  
1% 0.263184  
1% 0.0273438  
95% 94.7334  
1% 0.0273438  
1% 0.0273438  
18% 17.0591  
10% 9.77832  
1% 0.114746  
1% 0.0273438  
1% 0.0380859  
1% 0.100586  
7% 6.41602  
71% 70.8708  
1% 0.0273438  
1% 0.0273438  
1% 0.735352  
18% 17.3257  
27% 26.9341  
26% 25.5879  
1% 0.0273438  
9% 8.50537  
1% 0.027832  
2% 1.3833  
45% 44.9653  
1% 0.0273438  
49% 48.8945  
24% 23.4902  
15% 14.02  
20% 19.0938  
7% 6.72998  
1% 0.027832  
23% 22.3047  
1% 0.0273438  
1% 0.0273438

13% 12.2705  
1% 0.0273438  
1% 0.0292969  
1% 0.0273438  
1% 0.0273438  
15% 14.0386  
1% 0.0273438  
9% 8.80762  
1% 0.0273438  
42% 41.3147  
1% 0.964355  
1% 0.0273438  
1% 0.0273438  
12% 11.8242  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
1% 0.148926  
1% 0.034668  
15% 14.6084  
1% 0.0273438  
1% 0.0273438  
9% 8.33398  
1% 0.602539  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
6% 5.21826  
3% 2.28662  
20% 19.8179  
86% 85.3574  
13% 12.5635  
1% 0.0273438  
17% 16.5737  
1% 0.0273438  
1% 0.0292969  
5% 4.18262  
15% 14.8931  
4% 3.59814  
1% 0.0273438  
1% 0.0273438  
2% 1.22461  
10% 9.06055  
14% 13.4468  
11% 10.0293  
1% 0.0454102  
2% 1.25635  
9% 8.00977  
2% 1.37305  
1% 0.0273438  
1% 0.950684  
1% 0.861816  
1% 0.195801

1% 0.0273438  
1% 0.027832  
8% 7.41406  
1% 0.0727539  
1% 0.0273438  
1% 0.0273438  
1% 0.132812  
1% 0.674805  
1% 0.0273438  
1% 0.0649414  
1% 0.146484  
1% 0.0273438  
29% 28.2603  
45% 44.0356  
1% 0.0273438  
1% 0.0273438  
47% 46.104  
1% 0.139648  
1% 0.0742188  
1% 0.0273438  
8% 7.4585  
1% 0.0273438  
11% 10.2422  
1% 0.772949  
1% 0.027832  
1% 0.0273438  
1% 0.0273438  
1% 0.271484  
15% 14.4531  
3% 2.54443  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
23% 22.0583  
1% 0.222656  
1% 0.507812  
33% 32.5864  
1% 0.0288086  
1% 0.0273438  
14% 13.6055  
1% 0.0498047  
1% 0.0288086  
3% 2.9082  
13% 12.7832  
1% 0.0991211  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
3% 2.79395  
7% 6.1084  
42% 41.6829  
1% 0.0366211  
86% 85.5117  
1% 0.0273438



1% 0.0273438  
1% 0.0898438  
83% 82.1725  
9% 8.19189  
1% 0.0273438  
10% 9.03223  
1% 0.0273438  
22% 21.5479  
14% 13.627  
1% 0.0273438  
1% 0.0273438  
1% 0.201172  
1% 0.0541992  
1% 0.027832  
1% 0.0273438  
6% 5.02686  
1% 0.0273438  
1% 0.0273438  
10% 9.66943  
1% 0.0273438  
1% 0.027832  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
1% 0.443359  
13% 12.6255  
15% 14.4307  
1% 0.623047  
1% 0.109863  
1% 0.0874023  
30% 29.3188  
1% 0.0288086  
1% 0.0273438  
26% 25.043  
1% 0.399414  
12% 11.4507  
1% 0.0273438  
1% 0.0273438  
1% 0.161621  
1% 0.0273438  
19% 18.2129  
1% 0.902344  
1% 0.0288086  
1% 0.0273438  
1% 0.608398  
1% 0.769531  
4% 3.18213  
1% 0.647949  
95% 94.9639  
1% 0.0273438  
1% 0.119141  
5% 4.43457  
1% 0.518555  
1% 0.0273438

1% 0.0273438  
5% 4.74658  
1% 0.0273438  
1% 0.0859375  
6% 5.99561  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
14% 13.3335  
1% 0.0273438  
84% 83.6553  
1% 0.421387  
1% 0.0273438  
18% 17.1548  
1% 0.0273438  
1% 0.117676  
1% 0.0273438  
2% 1.2749  
1% 0.856934  
1% 0.0273438  
1% 0.129395  
1% 0.0273438  
20% 19.1455  
1% 0.0327148  
28% 27.6953  
4% 3.71484  
15% 14.8281  
9% 8.65625  
7% 6.37988  
49% 48.9609  
11% 10.6938  
1% 0.0273438  
1% 0.0717773  
1% 0.860352  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
32% 31.25  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
10% 9.72412  
1% 0.0273438  
1% 0.401367  
7% 6.36816  
1% 0.0273438  
1% 0.0273438  
1% 0.121094  
1% 0.0273438  
1% 0.027832  
1% 0.0273438

7% 6.98633  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
19% 18.332  
1% 0.0273438  
12% 11.6514  
1% 0.0292969  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
21% 20.6172  
1% 0.0273438  
1% 0.237793  
1% 0.0273438  
25% 24.4561  
61% 60.1363  
1% 0.291992  
12% 11.1187  
5% 4.14453  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
99% 98.543  
1% 0.106934  
16% 15.2173  
1% 0.0273438  
1% 0.0273438  
20% 19.9355  
1% 0.0576172  
1% 0.0273438  
1% 0.0273438  
67% 66.7129  
1% 0.0493164  
1% 0.0273438  
1% 0.0273438  
1% 0.529785  
1% 0.0668945  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
22% 21.6743  
1% 0.0273438  
1% 0.0273438  
3% 2.15967  
3% 2.80469  
1% 0.0273438  
41% 40.7422  
21% 20.3042  
1% 0.0273438  
1% 0.0854492  
2% 1.09814  
1% 0.102051

1% 0.0273438  
1% 0.0302734  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
1% 0.0571289  
1% 0.0610352  
1% 0.0273438  
25% 24.3555  
1% 0.0273438  
16% 15.5444  
1% 0.0283203  
13% 12.8892  
1% 0.0273438  
65% 64.6211  
9% 8.97363  
19% 18.8398  
16% 15.8525  
1% 0.0273438  
1% 0.0273438  
80% 79.876  
1% 0.0273438  
21% 20.1934  
1% 0.0273438  
1% 0.0273438  
1% 0.0317383  
1% 0.0273438  
2% 1.21045  
1% 0.0297852  
1% 0.0273438  
62% 61.5732  
1% 0.196289  
1% 0.110352  
1% 0.0273438  
8% 7.7666  
1% 0.0273438  
15% 14.0571  
1% 0.0273438  
1% 0.0473633  
16% 15.2173  
1% 0.0288086  
17% 16.5806  
1% 0.162109  
41% 40.5801  
9% 8.47559  
31% 30.916  
1% 0.0273438  
1% 0.0273438  
23% 22.4106  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
1% 0.0776367  
1% 0.0273438

1% 0.0273438  
1% 0.0273438  
19% 18.207  
2% 1.56592  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
1% 0.050293  
1% 0.0273438  
56% 55.7866  
20% 19.8223  
78% 77.0718  
1% 0.0273438  
1% 0.0273438  
1% 0.457031  
1% 0.931152  
4% 3.3208  
1% 0.0273438  
29% 28.4834  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
12% 11.8413  
1% 0.0273438  
11% 10.4399  
1% 0.0273438  
1% 0.0317383  
1% 0.0273438  
1% 0.0273438  
1% 0.107422  
1% 0.0273438  
38% 37.7568  
1% 0.57666  
1% 0.0273438  
1% 0.0273438  
1% 0.0307617  
1% 0.0273438  
1% 0.0273438  
3% 2.16162  
1% 0.0820312  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
1% 0.0273438  
24% 23.6367  
1% 0.181152  
2% 1.33545  
42% 41.8525  
1% 0.0273438  
20% 19.8579  
1% 0.0273438

3% 2.68115  
1% 0.458496  
31% 30.6118  
1% 0.0273438  
20% 19.5029  
1% 0.0273438  
1% 0.0273438  
1% 0.0297852  
21% 20.6763  
1% 0.0273438  
1% 0.0288086  
1% 0.0273438  
1% 0.0292969  
1% 0.0273438  
1% 0.0273438  
1% 0.027832  
1% 0.0634766  
1% 0.0288086  
1% 0.0288086  
1% 0.0288086  
37% 36.1255  
3% 2.29102  
13% 12.644  
1% 0.0273438  
2% 1.81689  
1% 0.0273438  
19% 18.8398  
4% 3.16553  
16% 15.71  
1% 0.0273438  
27% 26.9644  
5% 4.28564  
1% 0.0273438  
1% 0.0864258  
1% 0.0273438  
1% 0.0273438  
1% 0.0288086  
1% 0.0273438  
1% 0.414551  
23% 22.7393  
1% 0.0297852  
1% 0.0273438  
17% 16.7881  
1% 0.0351562  
1% 0.027832  
1% 0.73584  
1% 0.0273438  
29% 28.0127  
1% 0.0288086  
1% 0.0273438  
1% 0.0292969  
1% 0.0273438  
1% 0.0273438  
6% 5.11523









1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
41% 40.7056  
1% 0.0507812  
14% 13.5244  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0488281  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0273438  
1% 0.0507812  
1% 0.0507812  
1% 0.0273438  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
14% 13.5723  
1% 0.0507812  
11% 10.626  
1% 0.0488281  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0273438  
1% 0.0507812  
1% 0.0273438  
1% 0.0507812  
1% 0.0273438  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
26% 25.5259  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812







1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0273438  
1% 0.0507812  
1% 0.0507812  
3% 2.45996  
1% 0.0507812  
1% 0.0273438  
5% 4.01172  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
7% 6.67378  
1% 0.0507812  
1% 0.0507812  
4% 3.69922  
1% 0.0507812  
1% 0.0273438  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
14% 13.7134  
1% 0.0273438  
1% 0.0507812  
1% 0.0507812  
62% 61.6104  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0273438

1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
41% 40.4917  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
7% 6.45264  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0273438  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
5% 4.18457  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
18% 17.1157  
1% 0.0507812  
11% 10.4531  
1% 0.0507812  
8% 7.1875  
1% 0.0273438  
91% 90.7773  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0507812  
1% 0.0273438  
1% 0.0507812

```

1% 0.0507812
1% 0.0507812
1% 0.0488281
1% 0.0507812
1% 0.0507812
1% 0.0507812
1% 0.0488281
1% 0.0507812
1% 0.0488281
1% 0.0507812
1% 0.0488281
1% 0.0488281
1% 0.0488281
1% 0.0488281
1% 0.0273438
1% 0.0488281
1% 0.0488281
1% 0.0507812
11% 10.0879

```

To avoid division by zero, you actually have to say

```
12$ df | tail +2 | awk '{print $5, $2 == 0 ? 0 : 100 * $3 / $2}'
```

### ▼ Homework 1.3: how many 512-byte blocks are held back?

Write a shellscript named **heldback** that will run **df**, remove the first line of output with **tail** as shown above, and then print three columns: the **Filesystem** and **512-blocks** columns output by **df**, and the number of 512-byte blocks held back on each disk. This number is

```
512-blocks - Used - Avail
```

What determines the number of 512-byte blocks held back?

▲

(5) The commas shown above will output the fields with a blank separating them. If you omit the commas, the fields will be output with nothing separating them. See pp. 116, 125–126.

```
1$ awk -F: '{print $1, $3}' /etc/passwd
yqd6919 16033
```

```
2$ awk -F: '{print $1 $3}' /etc/passwd
yqd691916033
```

(6) You can also **print** a "double-quoted" string. See **OFS** on p. 120.

```
3$ awk -F: '{print $1 "--" $3}' /etc/passwd
yqd6919--16033
```

```
4$ awk -F: '{print $1 "\t" $3}' /etc/passwd
yqd6919      16033
```

```
5$ awk -F: '{print "\"" $1 "\" " $3}' /etc/passwd
"yqd6919" 16033
```

```
6$ awk -F: '{print $1 "\\ " $3}' /etc/passwd
yqd6919\16033
```



```
7$ head -5 /etc/passwd
root:x:0:0:root:/root:/sbin/sh
kroot:x:0:0::/root:/usr/bin/ksh
daemon:x:1:1::/
bin:x:2:2::/usr/bin:
sys:x:3:3::/
```

```
#include <sys/types.h>
```

```
typedef struct {
    uid_t user_id;           /* third field in /etc/passwd */
    gid_t group_id;        /* fourth field in /etc/passwd */
    char *login_name;      /* first field in /etc/passwd */
} user;
```

```
user u[] = {
```

```
8$ awk -F: '{print "\t{" $3 " ",\t" $4 " ",\t\" \"$1 \"\"},\t/* \" $5 \" */}' \
    /etc/passwd
```

```
{0,    0,    "root"},    /* root */
{0,    0,    "kroot"},    /* */
{1,    1,    "daemon"},    /* */
{2,    2,    "bin"},    /* */
{3,    3,    "sys"},    /* */
```

```
9$ awk -F: '{print "<TR><TD>" $3 "</TD><TD>" $4 "</TD><TD>" $1 "</TD></TR>"}' \
    /etc/passwd
```

```
<TR><TD>0</TD><TD>0</TD><TD>root</TD></TR>
<TR><TD>0</TD><TD>0</TD><TD>kroot</TD></TR>
<TR><TD>1</TD><TD>1</TD><TD>daemon</TD></TR>
<TR><TD>2</TD><TD>2</TD><TD>bin</TD></TR>
<TR><TD>3</TD><TD>3</TD><TD>sys</TD></TR>
```

```
10$ awk -F: \
    '{print "<TR>\n<TD>" $3 "</TD>\n<TD>" $4 "</TD>\n<TD>" $1 "</TD>\n</TR>\n"}' \
    /etc/passwd
```

```

<TR>
<TD>0</TD>
<TD>0</TD>
<TD>root</TD>
</TR>

<TR>
<TD>0</TD>
<TD>0</TD>
<TD>kroot</TD>
</TR>

<TR>
<TD>1</TD>
<TD>1</TD>
<TD>daemon</TD>
</TR>

<TR>
<TD>2</TD>
<TD>2</TD>
<TD>bin</TD>
</TR>

<TR>
<TD>3</TD>
<TD>3</TD>
<TD>sys</TD>
</TR>

```

### A gateway

Put this file in your `~/public_html/cgi-bin` subdirectory. `chmod` the file to `rwxr-xr-x`. `chmod` your home, `~/public_html`, and `~/public_html/cgi-bin` directories to `rwxr-xr-x`.

```

1 #!/bin/sh
2 #This file is named users.
3
4 echo 'Content-type: text/html'
5 echo
6 echo '<HTML>'
7 echo '<HEAD>'
8 echo '<TITLE>All acf5.nyu.edu users</TITLE>'
9 echo '</HEAD>'
10
11 echo '<BODY>'
12 echo '<H1>All acf5.nyu.edu users</H1>'
13 echo '<TABLE BORDER>'
14
15 awk -F: \
16 '{print "<TR>\n<TD>" $3 "</TD>\n<TD>" $4 "</TD>\n<TD>" $1 "</TD>\n</TR>\n"}' \
17     /etc/passwd
18
19 echo '</TABLE>'

```

```
20 echo '</BODY>'
21 echo '</HTML>'
22
23 exit 0
```

In your home page `~/public_html/index.html`, say

Click

```
<A HREF = "http://acf5.nyu.edu/cgi-bin/cgiwrap/~abc1234/users">here</A>
to see a list of all the acf5.nyu.edu users.
```

#### ▼ Homework 1.4: make a gateway that lists the users logged into acf5.nyu.edu right now

Pipe the output of `who` or `who -M` or `finger` or `w` into `awk`. The first column of the table should be line numbers: 1, 2, 3, 4, 5, etc.



Most Unix machines (but not i5) have a `/usr/pub/ascii` file. i5 has only `man ascii`.

```
1$ tail -8 /usr/pub/ascii
| 40 @ | 41 A | 42 B | 43 C | 44 D | 45 E | 46 F | 47 G | |
| 48 H | 49 I | 4a J | 4b K | 4c L | 4d M | 4e N | 4f O |
| 50 P | 51 Q | 52 R | 53 S | 54 T | 55 U | 56 V | 57 W |
| 58 X | 59 Y | 5a Z | 5b [ | 5c \ | 5d ] | 5e ^ | 5f _ |
| 60 ` | 61 a | 62 b | 63 c | 64 d | 65 e | 66 f | 67 g |
| 68 h | 69 i | 6a j | 6b k | 6c l | 6d m | 6e n | 6f o |
| 70 p | 71 q | 72 r | 73 s | 74 t | 75 u | 76 v | 77 w |
| 78 x | 79 y | 7a z | 7b { | 7c | | 7d } | 7e ~ | 7f del |
```

```
#!/bin/sh
#This shellsript is named splitascii.

awk '{print $2, $3 "\n" $5, $6 "\n" $8, $9 "\n" $11, $12 "\n" $14, $15 "\n" $17, $18 "'
```

```
#!/bin/sh
#This shellsript is named splitascii.

awk '{print \
    $2, $3 "\n" \
    $5, $6 "\n" \
    $8, $9 "\n" \
    $11, $12 "\n" \
    $14, $15 "\n" \
    $17, $18 "\n" \
    $20, $21 "\n" \
    $23, $24}'
```

```
2$ tail -8 /usr/pub/ascii | splitascii | more
40 @
41 A
42 B
43 C
44 D
45 E
46 F
47 G
```

(7) **NR**, **NF**, and **length** are **awk** built-in variables whose values change automatically as each line is input; see the list on p. 120. For example,

```
3$ awk '{print NR, $0}'           Number the lines; just like cat -n
4$ awk '{print $0, NR}'         Write the number at the end of each line.

5$ awk '{print NR - 1, $0}'     Start the numbers from 0 instead of 1
6$ awk '{print NR + 999, $0}'  Start the numbers from 1000 instead of 1
7$ awk '{print 1001 - NR, $0}' Start the numbers from 1000 and go down

8$ awk '{print 10 * NR, $0}'    Number the lines by tens.
9$ awk '{print 1 + (NR - 1) % 10, $0}' 1 to 10, go back to 1, repeat
```

NR	NR-1	(NR-1)%10	1+(NR-1)%10
1	0	0	1
2	1	1	2
3	2	2	3
4	3	3	4
5	4	4	5
6	5	5	6
7	6	6	7
8	7	7	8
9	8	8	9
10	9	9	10
11	10	0	1
12	11	1	2
13	12	2	3
14	13	3	4

```
char *a[] = {
    "moe",
    "larry",
    "curly",
    /* etc. */
};
```

Cut out the lines containing the strings, filter them through the following command, and paste them back in.

```
awk '{print "\t" $1 "\t/* " NR - 1 " */"}'
```

```
"moe",      /* 0 */
"larry",    /* 1 */
"curly",    /* 2 */
```

```
#!/bin/sh
#Output the lines of input in order of decreasing length.
#Sample use: longest < /usr/dict/words

awk '{print length, $0}' | #Add a number to the start of each line.
sort -nr | #Sort in decreasing numerical order.
sed 's/^[1-9][0-9]* //' #Remove the leading number and blank.

exit 0
```

```
10$ longest < /usr/dict/websters | more
thyroparathyroidectomize
tetraiodophenolphthalein
scientificphilosophical
pathologicopsychological
formaldehydesulphoxylate
transubstantiationalist
thymolsulphonephthalein
```

```
#!/bin/sh
#List /home and its descendants, one per line, in order of
#increasing depth (i.e., distance from the root directory).

find /home -type d -print |
awk -F/ '{print NF, $0}' | #Add a number to the start of each line.
sort -n | #Sort in increasing numerical order.
sed 's/^[1-9][0-9]* //' #Remove the leading number and blank.

exit 0
```

```
/home
/home/atar
/home/basson
/home/chapman
/home/froese
/home/konvit
/home/piemy
```

To prevent **find** from trying to search directories with no **r** and **x** permissions, the arguments of **find** actually have to be

```
find /home '(' ! -type d -o -perm -555 -o -prune ')' -type d -print
```

#### ▼ Homework 1.5: how long can a login name be?

**man -t 4 passwd** doesn't say how long a login name can be. Write a shellsript that outputs the length of the longest login name in the file **/etc/passwd**. The output of this shellsript should be a single line containing one number and nothing else. Use only one **awk**. Use **head**, not **tail**. On January 25, 2008, the longest login name had 8 characters.

You get no credit if you use `cat` or `<` in this shellsript. X52.9545 Handout 2, p. 24, ★ shows how to avoid them.

```
awk '{print length}'           number of characters in the entire input line
awk '{print length($1)}'      number of characters in only $1

sort                           alphabetical
sort -n                       ascending numeric (biggest at the bottom)
sort -nr                      descending numeric (biggest at the top)
```

Also write another shellsript using `uniq -c` and only one `awk` to output a table showing how many login names there are of each length, in order of increasing length. Do a numerical `sort`, not an alphabetical.

```
1 2
49 3
153 4
897 5
3272 6
109 7
12 8
```

```
1$ awk -F: 'length($1) >= 8 {print $1}' /etc/passwd | sort
barrem01
batal02
changj06
garcir02
hanleg01
lopezr04
noaccess
postgres
primas01
sherrs01
tokudm01
webservd
```



(8) Just as `$1`, `$2`, `$3` are the first, second, and third field on each line, `$NF` is the last field because `NF` is the number of fields.

```
2$ awk '{print NF}'           Print the number of fields on each line.
3$ awk '{print $NF}'         Print the last field on each line.
4$ awk '{print $NF - 1}'     Subtract 1 from the last field and print the remainder.
5$ awk '{print $(NF - 1)}'   Print the next-to-last field
6$ awk '{print $(NF - 2)}'   Print the second-from-last field
```

```
1 char a[] = "hello";
2 char *p = a + 1;          /* Let p point to the letter e. */
3
4 putchar(*p);              /* Print the letter e. */
5 putchar(*p - 1);         /* Print the letter d (1 less than e). */
6 putchar(*(p - 1));       /* Print the letter h (1 to left of e). */
7 putchar(p[-1]);          /* A better way to print the letter h in C. */
```

```
7$ lpq
Rank  Pri Owner      Job  Files                Total Size
1st   0   lql8156    569  ftp-rfc              579 bytes
2nd   0   jks2557    572  test                 21 bytes
3rd   0   iqs1378    579  mbox                 3764 bytes
4th   0   jwb8538    578  (standard input)    227090 bytes
```

```
8$ lpq | tail +2 | awk '{print $(NF-1)}'          Why not {print $6}?
579
21
3764
227090
```

Suppose you have ten rows and ten columns of numbers, i.e., a square matrix:

```
awk '{print $1}'          Print the first column.
awk '{print $NF}'        Print the last column.
awk '{print $NR}'        Print the main diagonal (upper left to lower right).
awk '{print $(NF - NR + 1)}' Print the other diagonal (upper right to lower left).
```

(9) You can print part of a string with the **substr** function. See p. 117.

```
9$ ls -l ~mm64/public_html/x52.9545/bio | tail +2
-r--r--r--  1 db91      users      351 Jun  5  2007 brown
-rw-----  1 ab2889   users      212 Jun  3  2007 bueno2
-r--r--r--  1 wob206   users      230 Oct 22  22:48 byrd
-r--r--r--  1 eucl       users      272 Nov  7  19:36 cates

10$ ls -l ~mm64/public_html/x52.9545/bio | tail +2 | awk '{print substr($0, 52, 2)}'
7
7
48
36

11$ ls -l ~mm64/public_html/x52.9545/bio | tail +2 | awk '{print substr($8, 4, 2)}'
12$ ls -l ~mm64/public_html/x52.9545/bio | tail +2 | awk '{print substr($8, 4)}'
13$ ls -l ~mm64/public_html/x52.9545/bio | tail +2 | cut -c52-53
```

(10) In addition to **print**, **awk** also has the C **printf** function. Write a **\n** at the end of the first argument and commas between the arguments, but don't write the parentheses.

```
14$ prog
111 10038 rpc206 Ruben Collins
100 09000 ag1917 Aliosha Gonzalez

15$ prog | \
awk '{printf "%03d %16s %-12s %05d %03d (%s)\n", NR, $5, $4, $2, $1, $3}'
001          Collins Ruben          10038 111 (rpc206)
002          Gonzalez Aliosha       09000 100 (ag1917)
```

```
16$ prog | \
awk '{printf "%03d %16s %-12s %05d %03d (%s)\n\n", NR, $5, $4, $2, $1, $3}'
001          Collins Ruben          10038 111 (rpc206)

002          Gonzalez Aliosha      09000 100 (ag1917)
```

```
17$ prog | awk '{printf "%s %s\n%d %d\n\n", $4, $5, $1, $2}'
Ruben Collins
111 10038

Aliosha Gonzalez
100 9000
```

### Split a long print onto two separate lines

Within the {curly braces} of an action, you need a backslash to split a `print` or `printf` onto two separate lines:

```
#!/bin/sh

awk '{print $1, $2, \
      $3, $4}'

awk '{printf "%5d %5d %5d\n", \
      $1, $2, $3}'
```

To split a double-quoted string onto two separate lines,

```
#!/bin/sh

awk '{print "supercalifragilisticexpialidocious"}'
awk '{print "supercalifragilistic" "expialidocious"}'
awk '{print "supercalifragilistic" \
      "expialidocious"}'
```

### Use awk to write the map file for a touch-sensitive image

The map file is `$mer/public_html/x52.9546/fa08_001.map`. It runs the gateway `$mer/public_html/cgi-bin/classphoto`.

```
aqp9925 254 75
akl201 193 113
ccl1 344 108
gho6432 272 105
jhv1 352 49
```

```
#!/bin/sh

awk '{printf "circle\t/cgi-bin/cgiwrap/~mm64/classphoto?%s\t%d,%d %d,%d\n", \
      $1, $2, $3, $2, $3 + 10}'
```



```
circle /cgi-bin/cgiwrap/~mm64/classphoto?aqp9925 254,75 254,85
circle /cgi-bin/cgiwrap/~mm64/classphoto?akl201 193,113 193,123
circle /cgi-bin/cgiwrap/~mm64/classphoto?cc11 344,108 344,118
circle /cgi-bin/cgiwrap/~mm64/classphoto?gho6432 272,105 272,115
circle /cgi-bin/cgiwrap/~mm64/classphoto?jhv1 352,49 352,59
```

### ▼ Homework 1.6: print a sample of each PostScript font

The fonts Symbol-Slanted and ZapfDingbats have no letters.

```
1$ cd/opt/sfw/share/groff/font/devps
2$ ls ?? | grep -v SS | grep -v ZD | head -3
3$ ls ?? | egrep -v 'SS|ZD' | head -3
AB
AI
AR

4$ grep '^internalname' `ls ?? | egrep -v 'SS|ZD'` | head -3
AB:internalname AvantGarde-Demi
AI:internalname AvantGarde-BookOblique
AR:internalname AvantGarde-Book

5$ grep '^internalname' `ls ?? | egrep -v 'SS|ZD'` | tr ':' ' ' | head -3
AB internalname AvantGarde-Demi
AI internalname AvantGarde-BookOblique
AR internalname AvantGarde-Book
```

*p. 28 for ??*

*p. 104 for |*

Write a shellscript named **fonts** starting with the commands shown above. The **grep** should output 18 lines. Then pipe the output of the **sed** into an **awk** containing exactly one **printf** to output 36 lines beginning with the six shown below. There must be no space before the **.br**; see p. 300. Use a double backslash to **print** a backslash. Be sure the **P** at the end of each sentence is uppercase (p. 299, Table 9.3). Use only one **awk**. There is no need for a second **cd** at the end of the shellscript.

### 6\$ fonts

Hand in the above output, and also the printout produced by

```
7$ fonts | groff -M/usr/lib/tmac -ms | lpr -Pedlab
8$ lpq -Pedlab
```

```
9$ fonts | groff -M/usr/lib/tmac -ms > file.ps
```

*If you have set up .faxrc,*

```
10$ fonts | groff -M/usr/lib/tmac -ms | $mer/bin/postfax
```

1. AvantGarde-Demi: **Pack my box with five dozen liquor jugs.**
2. AvantGarde-BookOblique: *Pack my box with five dozen liquor jugs.*
3. AvantGarde-Book: Pack my box with five dozen liquor jugs.



### The structure of each line of an awk argument: p. 114

Each line of an **awk** argument has one of the following three forms. The part in front of the curly brackets is called the *pattern*; the part within the brackets is called the *action*. You can omit either the pattern or the action from a line, but not both.

```

1$ awk '{print $3}'           Print the third field of every line.
2$ awk 'NR == 2 {print $3}'  Print the 3rd field of line 2 only; see p. 120 for ==. 'pattern {action}'

3$ awk 'NR == 2 {print $0}'  Print all of line 2 only.
4$ awk 'NR == 2 {print}'     Print all of line 2 only: p. 114.
5$ awk 'NR == 2'            Print all of line 2 only: p. 114.
6$ awk 'NR == 139069' $S45/Shakespeare.complete

7$ date
Friday, January 25, 2008 12:17:51 PM EST

8$ date | awk '{print $2}'   prints January
9$ cal | awk 'NR == 1 {print $1}' prints January

```

Even if `nslookup` is not in any of the directories on our `$PATH`, we can run it anyway if we tell the computer where to find it:

```

10$ /usr/sbin/nslookup i5.nyu.edu | awk 'NR == 5 {print $NF}'
128.122.253.152

11$ awk -F: '$4 == 15' /etc/passwd | more           variable doesn't have to be NR
shc329:x:17180:15:So Hyun Cho:/home1/s/shc329:/bin/ksh

See Handout 3, p. 25, for the comma:

12$ awk -F: '$4 == 15 {print $1, $5}' /etc/passwd | more
shc329 So Hyun Cho

13$ awk -F: '$4 == 15 {print $1 $5}' /etc/passwd | more
shc329So Hyun Cho

14$ cd /opt/sfw/bin
15$ ls -l | tail +2 | awk '$2 > 1' | more           files with multiple hard links
-r-xr-xr-x  2 root      bin          6551000 Apr 27  2007 emacs
-r-xr-xr-x  2 root      bin          6551000 Apr 27  2007 emacs-21.3

16$ grep '.....'
17$ awk 'length >= 60'           Print every line whose length is 60 or more characters.
18$ awk 'length >= 60 {print NR}' Print each line number instead of the line itself.
19$ awk 'length >= 60 {print NR, length}'

```

### Patterns with “and”, “or”, and regular expressions

```

1$ awk '10 <= NR && NR <= 20'           p. 120 for <= and &&
2$ awk '127702 <= NR && NR <= 127715' $S45/Shakespeare.complete

3$ ls -l | tail +2 | grep '^-' | awk '1000 <= $5 && $5 < 2000 {print $NF}'
4$ ls -l | awk 'NR >= 2 && /^-/ && 1000 <= $5 && $5 < 2000 {print $NF}'

5$ /usr/sbin/nslookup 128.122.253.152 | awk '/name = / {print $NF}'
15.NYU.EDU.

```

Because of the `$1 ~`, the `$` in `/\.ssh$` means the end of `$1`, not the end of the line.

```
6$ netstat -a -f inet -P tcp | awk '2 <= NR && NR <= 4 || $1 ~ /\.ssh$/' | head -6
TCP: IPv4
```

Local Address	Remote Address	Swind	Send-Q	Rwind	Recv-Q	State
*.ssh	*.*	0	0	65536	0	LISTEN
I5.NYU.EDU.ssh	ND-IMAC-03.NDLAB.ITS.NYU.EDU.60926	65535			0	66608 0 ESTAB
I5.NYU.EDU.ssh	ND-IMAC-03.NDLAB.ITS.NYU.EDU.61109	65535			0	66608 0 ESTAB

### When not to use awk

A special-purpose tool such as **head** is faster and simpler than a multi-purpose, programmable tool such as **awk**.

```
1$ awk 'NR == 1'           The fox knows many little things; the hedgehog knows one big thing.
2$ head -1                faster and simpler way to do the same thing

3$ awk 'NR <= 10'
4$ head -10               faster and simpler way to do the same thing
5$ head                   even simpler: -10 is the default (Handout 3, p. 25)

6$ awk 'NR >= 13'
7$ tail +13               faster and simpler way to do the same thing
```

### Patterns that compare two strings

Like many languages, awk requires "double quotes" around a string.

```
1$ awk -F: '$4 == 15 {print $1}' /etc/passwd | more
2$ awk -F: '$7 == "/bin/ksh" {print $1}' /etc/passwd | more

3$ awk -F: '$7 != "/bin/ksh" {print $7}' /etc/passwd | sort | uniq
/bin/bash
/bin/csh
/bin/false
/bin/tcsh
/sbin/sh
/usr/bin/bash
/usr/bin/csh
/usr/bin/ksh
/usr/bin/pfksh
/usr/bin/tcsh
/usr/lib/uucp/uucico
/usr/local/etc/expiredshell

4$ awk -F: '$7 != "/bin/ksh" && $7 != "" {print $7}' /etc/passwd | sort -u

5$ awk '$1 == "UserDir" {print $2}' /etc/apache/httpd.conf
public_html
```

```
6$ cd /dev/pts On other systems, /dev or /devices/pseudo
7$ pwd
8$ ls -l | awk 'NR >= 2 && $3 != "root"' | head -5
crw--w---- 1 jsb3      tty      24,  1 Jan  7 10:01 1
crw--w---- 1 mm64      tty      24, 10 Jan 25 11:28 10
crw--w---- 1 puskar     tty      24, 11 Jan 24 17:26 11
crw--w---- 1 bjr270     tty      24, 12 Jan 24 16:46 12
crw--w---- 1 bjr270     tty      24, 13 Jan 24 16:46 13
```

Sort the output of `awk` into increasing numeric order, ignoring the first 9 fields on each line. See p. 106.

```
9$ ls -l | awk 'NR >= 2 && $3 != "root"' | sort +9n | head -5
crw--w---- 1 jsb3      tty      24,  1 Jan  7 10:01 1
crw--w---- 1 bjr270     tty      24,  2 Jan 22 15:34 2
crw--w---- 1 bjr270     tty      24,  4 Jan 22 15:34 4
crw--w---- 1 bjr270     tty      24,  5 Jan 22 15:35 5
crw--w---- 1 mm64      tty      24,  6 Jan 25 12:17 6
```

```
10$ grep '::' /etc/passwd
11$ awk -F: '$2 == ""' /etc/passwd people with no password: pp. 116–117
```

```
#!/bin/ksh
#Output the last name of everyone in the class, one per line.

for file in ~mm64/public_html/x52.9545/bio/*
do
    awk 'NR == 2 {print $NF}' $file
done

exit 0
```

```
Brown
Byrd
cates
Chung
```

```
#!/bin/ksh
#Output the home phone of every student in the class, one per line.

for file in ~mm64/public_html/x52.9545/bio/*
do
    awk 'NR == 5' $file
done

exit 0
```

```
718.757-8484
609-555-2929
home phone (212) 732-3075
Home: 555-555-5554
```

Output the names of the directories at the third level (e.g., `/usr/dict`, `/usr/games`, `/home1/a`, `/home1/m`, etc.):

```
12$ find / -type d -print | awk -F/ 'NF == 3' > ~/temp &
13$ find / -type d -print | grep '^[^/]*/[^/]*$' > ~/temp &
```

To prevent `find` from trying to search directories with no `r` and `x` permissions, the arguments of `find` actually have to be

```
find / '(' -type d -o -perm -555 -o -prune ')' -type d -print
```

```
14$ ps -Aj | head -5
      PID  PGID   SID  TTY          TIME CMD
      525   525   525  ?           0:00 zsched
    13339 13336 13234  ?           0:00 bash
      667   525   525  ?           0:04 init
    13272 13272 13234  ?           0:00 bash
```

```
#!/bin/sh
#Output the login name of everyone running more than 30 programs.
#Instead of printing them in alphabetical order, could you print them
#in increasing order of how many programs they were running?
```

```
ps -Aj |
awk 'NR >= 2 {print $1}' |
sort |
uniq -c |
awk '$1 > 30 {print $2}'
```

```
exit 0
```

```
#!/bin/sh
#Kill all programs not owned by root.
#Only the superuser can run this shellscript.
```

```
kill -9 `ps -Aj | awk 'NR >= 2 && $1 != "root" {print $2}``
```

```
exit 0
```

### Search a `.c` file's symbol table for names that are too long

A variable or function that can be used in more than one `.c` file of a C program is called a *global*. On some machines the name of a global can be no more than six characters. See the K&R C book, p. 192, §A.23.

```
#include <stdio.h>
```

```
int globalvar1;
int gv2;
void f(void);
```

```
main()
{
    f();
}
```

```
1$ gcc -c file.c           Create file.o
2$ nm -g file.o           List the names of the globals.
```

```
10427.o:
```

[Index]	Value	Size	Type	Bind	Other	Shndx	Name
[7]					4	4	OBJT   GLOB   0   COMMON
[8]					4	4	OBJT   GLOB   0   COMMON
[6]					0	16	FUNC   GLOB   0   1

```
3$ nm -g file.o | awk 'length($1) > 6 {print $1}'
10427.o:
[Index]
```

### ▼ Homework 1.7: print loginnames longer than 7 characters

Write a shellscript named `longlogs` that will output all the loginnames in `/etc/passwd` that have more than seven characters. The shellscript must contain exactly one `awk` and nothing else.

```
1$ longlogs
webservd
postgres
noaccess
barrem01
batal02
changj06
garcir02
hanleg01
lopezr04
primas01
sherrs01
tokudm01
```

▲

### Regular expressions in an awk pattern: pp. 117, 119

```
awk 'NR == 2 {print $3}'
awk '/^moe/ {print $3}'
awk '!/^moe/ {print $3}'
```

#Output the name of every file in the current directory.

```
ls -l | tail +2 | grep '^-' | awk '{print $NF}'
ls -l | tail +2 | awk '/^-' {print $NF}'
```

#Output the name of everything in the current directory whose owner's login name is of the form abc1234. See p. 117 for the tilde ~.

```
ls -l | tail +2 | awk '$3 ~ /^[a-z][a-z][a-z][0-9][0-9][0-9][0-9]$/ {print $NF}'
```

#Output the name of everything in the current directory

#whose owner's login name is not of the form abc1234 (exclamation tilde).

```
ls -l | tail +2 | awk '$3 !~ /^[a-z][a-z][a-z][0-9][0-9][0-9][0-9]$/ {print $NF}'
```

```
#man 4 passwd says that a login name can't contain an uppercase
#letter. Is this true?
awk -F: '$1 ~ /[A-Z]/' /etc/passwd
grep '^[^:]*[A-Z]' /etc/passwd harder way to do same thing
```

```
#Output the first field of every line that contains the fraction 1/2.
awk '/1\|2/ {print $1}'
```

List all the courses given this semester on i5,

```
1$ awk -F: '$1 ~ /083:/' /etc/group
```

### Regular expression example

```
1$ awk -F: '$6 ~ /^\/home\/' {print $1}' /etc/passwd | head -3
kraskey
konvit
atar
```

```
2$ mail `awk -F: '$6 ~ /^\/home\/' {print $1}' /etc/passwd` < ~/letter
```

### awk and egrep

The regular expressions you write between the // in **awk** is the same as the regular expression you write in **egrep**:

```
grep '\(.\)\|1' grep can do this, but egrep can't
egrep 'pro(choice|life)' egrep can do this, but grep can't

awk 'NR < 10 || NR > 20'
awk '/prochoice/ || /prolife/' never write this
awk '/prochoice|prolife/' easier way to do the same thing
awk '/pro(choice|life)/' What would go wrong without the parentheses?
```

### Regular expression vs. test for equality

```
#Print the first field of every line whose third field is moe. ☞
awk '$3 == "moe" {print $1}'

#Print the first field of every line whose third field contains moe.
awk '$3 ~ /moe/ {print $1}'

#Print the first field of every line whose third field begins with moe.
awk '$3 ~ /^moe/ {print $1}'

#Never write this: it's faster to write ☞ instead.
awk '$3 ~ /^moe$/ {print $1}'
```

### Search for a regular expression in /etc/group

```
1$ awk -F: '$1 == "x529545001" {print $1}' /etc/group | head -5
```

```
2$ awk -F: '$1 ~ /x529545001/ {print $1}' /etc/group | head -5
x52954500120063
x52954500120071
x52954500120072
x52954500120073
```

```
3$ awk -F: '$4 == "mm64" {print $1}' /etc/group | head -5
```

```
4$ awk -F: '$4 ~ /mm64/' /etc/group | head -5
```

```
5$ awk -F: '$4 ~ /(^\|,)mm64(,|$)/' /etc/group | head -5
```

### List the running programs that are not shells

```
1$ w | more
 12:17pm up 22 day(s), 20:49, 3 users, load average: 0.18, 0.10, 0.09
User      tty          login@ idle   JCPU   PCPU   what
kroot     pts/7         Thu 5pm 38     1     -ksh
mm64     pts/6         9:18am  2:12   w
mm64     pts/8         9:10am  4     1:51   vi /tmp/Re9024
```

```
2$ awk 'NR >= 3 && substr($0, 67) !~ /^-sh$|^-csh$|^-ksh$/' | more
```

```
3$ awk 'NR >= 3 && substr($0, 67) !~ /^-(sh|csh|ksh)$/' | more
```

```
kroot     pts/7         Thu 5pm 38     1     -ksh
mm64     pts/6         9:18am  2:12   w
mm64     pts/8         9:10am  4     1:51   vi /tmp/Re9024
```

### A shell variable inside the argument of awk: p. 126

In an argument of `awk`, `$1`, `$2`, `$3` stand for the fields on each line that is input to `awk`. In a shellsript, however, `$1`, `$2`, `$3` stand for the shellsript's command line arguments, except when enclosed in single quotes. That's why we enclose the argument of `awk` in single quotes.

```
other::1:root
users::15:
```

```
1$ grep '^[^:]*:[^:]*:15:' /etc/group
```

```
#!/bin/sh
#Output the name of the group whose GID number is 6429.

awk -F: '$3 == 6429 {print $1}' /etc/group
exit 0
```



```
#!/bin/sh
#Output the name of the group whose GID number is given as a command
#line argument.  Sample use:
#   2$ groupname 6429

if [ $# -ne 1 ]
then
    echo $0: requires one argument giving the group number. 1>&2
    exit 1
fi

awk -F: '$3 == '$1' {print $1}' /etc/group
exit 0
```

```
#!/bin/sh
#Output the GID number of the group whose name is x52954600120083.
#Like most programming languages, the awk language requires "quotes"
#around a string.

awk -F: '$1 == "x52954600120083" {print $3}' /etc/group
exit 0
```

```
#!/bin/sh
#Output the GID number of the group whose name is given as a command
#line argument.  Sample use:
#   3$ groupnumber x52954600120083

if [ $# -ne 1 ]
then
    echo $0: requires one argument giving the group name. 1>&2
    exit 1
fi

awk -F: '$1 == "'$1'" {print $3}' /etc/group
exit 0
```

```
#!/bin/sh
#Output the name of everything in the current directory that is not
#owned by the person who is running this shellscript.
#See man 5 environ for the $USER variable.

ls -l | tail +2 | awk '$3 != "'$USER'" {print $NF}'
```

#### ▼ Homework 1.8: output a UID number

Write a shellscript named `uid` that will output the **UID** number of the user whose login name is given as a command line argument. Your **UID** number is the third field of your line in the file `/etc/passwd`. Use the method shown above. Sample use:

```
1$ uid aa1272
14836
```



### ▼ Homework 1.9: output someone's PID numbers

Write a shellscript named `pid` that outputs the **PID** numbers, one per line, of all of the processes owned by the user whose login name is given as a command line argument. Start with `ps -Af` and use the method shown above. Sample use:

```
1$ pid aa1272
100
200
300
```

▲

### ▼ Homework 1.10: kill all of someone's processes

Write a shellscript named `killall` that will **kill -9** all the processes owned by the user whose login name is given as a command line argument. For example, to **kill** all of `abc1234`'s processes, the superuser would type

```
1$ killall abc1234
```

`killall` can run the previous homework's `pid` shellscript in back quotes.

You should never attempt to kill the process whose PID is 0; see `man -t kill | grops | lpr`. Is it simpler to use `grep -v` or `awk '$1 > 0'` to remove 0 from the output of the previous homework?

Test `killall` by trying it on someone else. All you should get are error messages saying **Not owner** because Unix won't let you **kill** other people's processes.

▲

### Awk example: log off all the idlers

If a line input to `awk` has less than eight fields, then `$8` will contain the null string. In an `awk` regular expression, `?` means "zero or one" and `|` means "or".

```
1$ finger -i
```

Login	TTY	When	Idle
kroot	pts/7	Thu Jan 24 17:47	37 minutes
mm64	pts/8	Fri Jan 25 09:10	3 minutes 30 seconds
mm64	pts/6	Fri Jan 25 09:18	15 seconds

```
1 finger -i | tail +2 | awk '$8 == "hour" || $8 == "hours"'
2 finger -i | tail +2 | awk '$8 ~ /^hours?$/ '
3 finger -i | tail +2 | awk '$8 ~ /^hours?$/ || $8 ~ /^days?$/ '
4 finger -i | tail +2 | awk '$8 ~ /^hours?$/|^days?$/ '
5 finger -i | tail +2 | awk '$8 ~ /^(hour|day)s?$/ '
```

```
#!/bin/sh
#This shellscript is named terminate.
#Output the name of every terminal which has been idle for one hour or
#more. Sample use: 2$ terminate

finger -i |
awk 'NR >= 2 && $2 != "console" && $8 ~ /^(hour|day)s?$/ {print $2}' |
sed 's/^\*// '

exit 0
```

```
3$ terminate
```

```
4$ ps -A -otty,pid | head -5
TT      PID
?       525
?       13339
?       667
?       13272
```

Pretend that terminals `ttyp1`, `ttyp2`, and `ttyp3` were the three idle terminals that we want to log out. Command 1 would output the PID number of each program running on terminal `ttyp1`. Command 2 would kill each program running on terminal `ttyp1`:

```
1 5$ ps -A -otty,pid | awk 'NR >= 2 && $1 == "ttyp1" {print $2}'
2 6$ kill -9 `ps -A -otty,pid | awk 'NR >= 2 && $1 == "ttyp1" {print $2}`
3 7$ kill -9 `ps -A -otty,pid | awk 'NR >= 2 && $1 == "ttyp2" {print $2}`
4 8$ kill -9 `ps -A -otty,pid | awk 'NR >= 2 && $1 == "ttyp3" {print $2}`
5 9$ ps -A -otty,pid | tail +2 > ~/temp
6 10$ kill -9 `awk '$1 == "ttyp1" {print $2}' ~/temp`
7 11$ kill -9 `awk '$1 == "ttyp2" {print $2}' ~/temp`
8 12$ kill -9 `awk '$1 == "ttyp3" {print $2}' ~/temp`
```

```
#!/bin/ksh
#Kill every process running on a terminal which has been idle for one
#hour or more. See p. 96 for back quotes in the for line.

ps -A -otty,pid | tail +2 > ~/temp

for terminal in `terminate`
do
    kill -9 `awk '$1 == "$terminal" {print $2}' ~/temp`
done

rm ~/temp
exit 0
```

The superuser can run the above shellsript every 15 minutes by using `at`. You can rewrite it more efficiently with `join`.

### ▼ Homework 1.11: make the threshold 35 minutes

Change the `terminate` shellsript to output the name of every terminal except the console that has been idle for 35 minutes or longer.

Use `||` and `&&` (p. 120) in the argument of `awk`; `&&` has higher precedence. You get no credit if the argument of `awk` contains the number 34. Make sure your `awk` finds 5 of the 10 terminals in the file `$$S46/idle`.

You get no credit if you use `cat` or `<` in this shellsript. X52.9545 Handout 2, p. 24, ★ shows how to avoid them.

▲

### emacs for vi users

**Emacs** is not an acronym. It is named **emacs** because it is an editor that was originally written with a lot of macros.

**Print the 290-page GNU Emacs Manual (not required)**

After printing the manual, move the tables of contents to the front. Page numbers below refer to this manual.

```

1$ cd
2$ ftp gatekeeper.dec.com                Digital Equipment Company
Name (gatekeeper.dec.com:abc1234): anonymous
Password: abc1234@acf4.nyu.edu          your network address (invisible)
ftp> cd pub/GNU
ftp> binary                             emacs.dvi.Z is not an ASCII file.
ftp> get emacs.dvi.Z                    Be patient.
ftp> quit

3$ ls -l
-rw-r--r--  1 abc1234      372135 Sep 28 13:12 emacs.dvi.Z

4$ uncompress emacs.dvi.Z                or gunzip
5$ ls -l
-rw-r--r--  1 abc1234     1164620 Sep 28  emacs.dvi

6$ lpr -d emacs.dvi                      -d option to print a .dvi file
7$ rm emacs.dvi

```

**Create an emacs initialization file: pp. 232–235**

Use **vi** to create a file named **.emacs** in your home directory containing the Lisp functions that you want **emacs** to execute automatically when you invoke it.

```

;This file is $HOME/.emacs

;Display the time on the mode line (p. 16).
(display-time)

;Make searching case-sensitive (p. 78).
(setq-default case-fold-search nil)

;If cursor falls off bottom of window, scroll up only 1 line (p. 66).
(setq-default scroll-step 1)

```

**Insert mode and escape: pp. 17–18**

In **emacs**, there is no separate insert mode—you're always in insert mode. Everything you type gets inserted into the file except for control characters or meta characters (**control-x**, etc).

All **emacs** commands (move up, move down, delete, etc.) begin with one of these special characters. You already know how to type a control character such as **control-x**: press the control key and keep it down while you also press **x**. Instead of writing **control-x** we will abbreviate it **C-x**. Don't type the dash.

A meta character is just the same, except you press the **meta** (or **alt**) key instead of the control key. We abbreviate it **M-x**. If you have no meta key (or if it is reserved for other purposes as is the **alt** key on an IBM PC in LC-8 Tisch), press the **ESC** key, then release it, and then press the **x** key.

To summarize: **C-x** means to press **control** and then press **x** while the **control** key is still down. **M-x** means to press **ESC** and then press **x** after you have released the **ESC** key.

**Edit multiple files with vi and emacs**

When you invoke **vi**, you specify the name of the file to be edited as a command line argument. A **vi** user often runs several copies of **vi** simultaneously, each one editing a different file. You switch between them with **control-z**, **jobs**, **%1**, **%2**, etc. A common mistake is to run two **vi**'s on the same file at the same time.

```
1$ vi file1
control-z
2$ vi file2
control-z
3$ vi file3
control-z
4$ vi file1          Big mistake; should have typed %1 instead.

5$ jobs             See a list of the files you're editing.
```

This mistake never happens with **emacs**, however, because you run only one copy of **emacs** to edit many files simultaneously. Invoke **emacs** with no command line arguments (p. 22), and then specify the files to be edited. For example,

```
6$ emacs
C-x C-f file1 RET   Edit the file file1; create it if it doesn't exist (pp. 88-89).
C-x C-f file2 RET   Don't type the dashes or the spaces.
C-x C-f file3 RET
C-x C-f file1 RET   Go back to file1; don't start editing another copy.

C-x C-b             See a list of the files you're editing, p. 106.
C-x 1               Make the list disappear (the digit "one"), p. 114.
```

When you type **C-x C-f**, **emacs** will try to write the first part of the file's pathname for you, e.g., **~/**. If this is the wrong path, delete these characters with **M-0 C-k** (the digit zero) before you type the filename.

**A practice file: p. 45**

Instead of hacking up the file **\$\$45/poem**, practice on the file **/usr/local/gnuemacs/etc/TUTORIAL**. Because this is a special file, you can edit it by the "help-tutorial" command

```
C-h t              Don't type the dash or the space.
```

instead of by the normal

```
C-x C-f /usr/local/gnuemacs/etc/TUTORIAL RET
```

command shown above. **C-h t** will let you mangle a copy of the **TUTORIAL** file without damaging the original.

Before using **emacs** make sure you have said **ts vt100**, or put this command into your **.login** file.

```
1$ cd
2$ ts vt100         if this is not already in your .login file
3$ emacs
GNU Emacs 18.55.2 of Fri Oct 19 1990 on acf9 (berkeley-unix)
C-h t              Practice editing on the TUTORIAL file.
You are looking at the Emacs tutorial.
```

```

C-x C-c                               Exit from emacs when you've had enough (p. 22).
Save file /home1/a/TUTORIAL? (y or n) n
1 modified buffer exists, do you really want to exit? (yes or no) yes
4$

```

### Move the cursor: pp. 26–27

The four arrow keys will move the cursor on most terminals; hold the keys down to repeat. Otherwise, use

```

C-p          up          (“previous”)
C-n          down        (“next”)
C-b          left        (“back”)
C-f          right       (“forward”)

M-x goto-line RET      Go to line 25.
Goto line: 25 RET

M-<          Go to the first line in the file.
M->          Go to the last line in the file.
M-x what-line RET      See what line number the cursor is at now.

```

The **M-f** and **M-b** keys will move the cursor to the next (or previous) word. The **C-a** and **C-e** keys will move the cursor to the start (or end) of the line.

### Search: pp. 69–78

To move the cursor to the next occurrence of a string (e.g., “animal”),

```
C-s animal          incremental search
```

Press **C-s** again to search for the next animal, or **ESC** to terminate the search. Thus, to search for only the first animal, terminate the search immediately with an **ESC**:

```
C-s animal ESC
```

If the search fails (**Failing I-Search**) and you type **C-s** again, the search will wrap around to the start of the file.

If your terminal freezes when you type **C-s**, unfreeze it with **C-q**. Then type **M-x isearch-forward RET animal ESC**, and make a mental note to rebind the **isearch-forward** function to a different key as shown below.

**C-r** is the same as **C-s**, except that it searches backwards. **C-M-s** is the same as **C-s**, except that it searches for a regular expression:

```
C-M-s ^[Mm]oe ESC      Type ESC C-s before the ^[Mm]oe (p. 17).
```

To search backwards for a regular expression,

```
M-x isearch-backward-regexp RET
Regexp I-search backward: ^[Mm]oe ESC
```

### Scroll: p. 65

If the file is too big to fit on the screen at once, you can scroll the buffer by using the arrow keys to move the cursor off the top or bottom edge; see **scroll-step** in the **.emacs** file. Or

<b>C-v</b>	<i>Scroll down almost a full screen.</i>
<b>M-v</b>	<i>Scroll up almost a full screen.</i>
<b>C-u 4 C-v</b>	<i>Scroll down 4 lines. C-u is the universal repetition prefix.</i>
<b>C-u 4 M-v</b>	<i>Scroll up 4 lines.</i>

### Delete and kill: pp. 53–55

To delete a character, aim at it (i.e., place the cursor on it) and press **C-d**. This key repeats and will suck in text from the right. **C-u 4 C-d** will delete four characters. The rubout key (usually marked with a left arrow) deletes the character to the left of the cursor.

To append the following line to the current line, delete the invisible newline character at the end of the current line.

To kill a word, aim at the first letter of the word and type **M-d**. Or type **C-u 4 M-d** to kill four words.

To kill a line (or at least the part of the line starting at the cursor), type **C-k**. A second **C-k** will kill the newline as well. **C-u 4 C-k** will kill four lines.

To undo a kill (or anything else), type **C-x u**.

### Kill an entire region: pp. 49–50

To kill a large region of text, aim at the first character of the region and type **C-space bar** to mark the start of the region. The exact starting point is the left side of the cursor; **emacs** should say **Mark set**. (If not, type **M-x set-mark-command RET**, and make a mental note to rebind the **set-mark-command** function to a different key as shown below.) Then aim at the character after the last one you want to delete and type **C-w**.

You can also mark the end of the region with **C-space bar** and then **C-w** at the start of the region.

### Re-insert the killed text somewhere else: pp. 55–58

“Deleted” text is gone forever, but each chunk of killed text is stored in the “kill ring”. To insert this text back into your file wherever you wish, aim at the desired point and type **C-y** (for “yank”). You can re-insert the text any number of times, and even insert it into a different file (see “Edit several files” below).

**Vi** has two different re-insertion commands, **p** and **P**. In fact, it has several such pairs: **o** and **O**, **a** and **i**. None of this is necessary in **emacs**, however, because **emacs** always inserts the text at the point to the left of the cursor and you can move the cursor to the right of the last character on the line.

### Duplicate existing text

You can kill a body of text and then, without moving, re-insert it twice. This will restore the text and give you an extra copy. To duplicate a line, for example, go to the start of the line and type **C-k C-k C-y C-y**.

To copy a region into the kill ring without killing it (i.e., to paste instead of cut), mark the start of the region with **C-space bar**, travel to the end of the region, and type **M-w** instead of **C-w** (p. 56).

### Create new text: p. 25

Go to the point at which you want to insert new text and simply type it, pressing **RET** whenever you want to start a new line. There is no “insert mode” to get into and out of. The price you pay for this freedom is the unfortunate fact that all **emacs** commands must begin with **C-** or **M-**.

### Global substitution: p. 79

vi	emacs
<code>:.,\$s/moe/larry/g</code>	<code>M-x replace-regexp RET</code> <code>Replace regexp: moe RET</code> <code>with: larry RET</code>
<code>:1,\$s/moe/larry/g</code>	<code>C-&lt;</code> <code>M-x replace-regexp RET</code> <code>Replace regexp: moe RET</code> <code>with: larry RET</code>
<code>:10,20s/moe/larry/g</code>	<i>See pp. 79, 208, 229.</i>

### Edit several files at the same time: pp. 105–109

To edit several files simultaneously,

```
C-x C-f file1 RET      Create a buffer for file1 and start editing it.
C-x C-f file2 RET
C-x C-f file3 RET
C-x C-f file1 RET      Go back to the buffer for file1—don't create another.
```

Each file is edited in its own “buffer”, which is named after the file. In addition, there are other buffers for writing in Lisp, reading mail, talking to the doctor (p. 240), etc.

```
C-x C-b              Display a list of your buffers.
C-x 1                Remove the list; the digit one.
C-x b file1 RET      Go to the buffer named file1.
C-x k file1 RET      Kill the buffer named file1. It will ask if you want to save.
```

Once you have created a buffer for editing the file `file1`, the commands `C-x C-f file1 RET` and `C-x b file1 RET` do the same thing. `C-x b` often requires less typing, however, because it guesses which buffer you want to go to. If it guessed correctly, just press `RET` without typing the name of the buffer.

### Display several files at the same time: pp. 112–114

To cut and paste rapidly between different files, you can squeeze two buffers onto the screen simultaneously.

```
C-x C-f file1
C-x 4 C-f file2      Split the screen
C-x o                Move the cursor from file2 back to file1 (lowercase o).
C-x o                Move the cursor from file1 back to file2.
C-x ^                Enlarge the window where the cursor is.
```

Let’s say you are in the `file2` window and have finished editing it.

```
C-x C-s              Save the file; see below.
C-x 0                Remove the window; let other window expand (the digit zero).
```

`C-x 0` removes the window where the cursor is; `C-x 1` removes every window except the one where the cursor is (pp. 113–114).

### Write your changes back to the disk: p. 90

When you `C-x C-f` a file, `emacs` makes a copy of the file you are editing and displays the copy on the screen. All of your editing commands are applied only to the copy. The original version of the file on the disk remains unchanged until you give the “save” command `C-x C-s`. If your terminal freezes when you type `C-s`, unfreeze it with `C-q`. Then type `M-x save-buffer RET`, and make a mental note to rebind the `save-buffer` function to a different key as shown below.



**Exit from emacs: pp. 21–22**

**C-x C-c** will terminate **emacs**. If you haven't saved your file(s), **C-x C-c** will give you one last chance.

When you log in, invoke **emacs** once. When you need to do other things, stop **emacs** with **control-z** and restart it with **%1** (or **%2**, etc). Do not invoke another copy of **emacs**.

**What can go wrong**

To abort a command after typing the first characters, type **C-g** (pp. 35, 237).

If the **ESC** key doesn't work, try **control-[**.

If a "disabled command" window pops up asking you to

**Type y, n or Space:**

just press **n** without pressing **RET**. See p. 229.

To refresh the screen after someone tried to **write** or **talk** to you, type **C-l** (lowercase L), pp. 65–66.

**Rebind a function to a different key: pp. 19–20, 226–228**

Several **emacs** commands require you to type **C-s**: for example, **C-s** to search forward for a string, and **C-x C-s** to save a file to disk. On many terminals, however, **C-s** and **C-q** freeze and unfreeze the screen. These are not **emacs** commands—they are special features of the keyboard.

If you find that **C-s** freezes **emacs**, you can bind the search and save functions to different keys. For example, to bind search to **C-t** instead of **C-s**, and save to **C-x C-t** instead of **C-x C-s**, write the following in the **.emacs** file.

```
;Can't use C-s because it would freeze the screen.
(global-set-key "\C-t" 'isearch-forward)                ;p. 69
(global-set-key "\C-x\C-t" 'save-buffer)                ;p. 90
```

To bind the search and save functions to keys on the **vt100** numeric keypad, you must use the **SS3-map**. Unfortunately, this data structure is not created until after the **.emacs** file has finished executing (p. 235). The solution is to put the key binding instructions in a separate function at the end of the **.emacs** file, and arrange to have this function executed after the **SS3-map** is created. See p. 236 for this "hook".

```
;Write this at the end of your $HOME.emacs file.
```

```
(defun term-setup-hook-vt100 ()                ;Define a function with no args.
  "After executing the terminal-dependent initialization in the
  file /usr/local/gnuemacs/lisp/term/vt100.el, execute this
  function to redefine the numeric keypad keys listed in the
  vt100.el file."
  (define-key SS3-map "w" 'isearch-forward)                ;7 key
  (define-key SS3-map "P" 'save-buffer)                    ;PF1 key
  (define-key SS3-map "x" 'goto-line)                      ;8 key
  (define-key SS3-map "u" 'what-line)                     ;5 key
)
(setq term-setup-hook 'term-setup-hook-vt100)
```

**Display help: pp. 45–48**

To create a help window about a function, variable, or key,

**C-h f replace-regexp RET**            *a function such as **replace-regexp**.*  
**C-h v case-fold-search RET**        *a variable such as **case-fold-search**.*  
**C-h k C-x C-f**                        *a key such as **C-x C-f**.*  
**C-x 1**                                 *Remove the help window.*

To scroll, save, or enlarge the help window, move the cursor into it with the **C-x o** command (lowercase **o**) described above. For example, print the help window for the **interactive** function before attempting the Lisp programming below.

**C-h f interactive RET**    *a function such as **interactive**.*  
**C-x o**                                 *Move the cursor into the help window (lowercase **o**).*  
**C-x C-w junk RET**         *Save a copy in a file named **junk** to print later.*  
**C-x 0**                                 *Remove the help window (the digit zero).*  
**C-z**                                    *to stop **emacs***  
**Stopped**  
**1\$ pr -l60 -h interactive junk | lpr**                         *minus lowercase L sixty*  
**2\$ rm junk**  
**3\$ %1**                                 *or %2, etc., to restart **emacs***

**How to call a function in Lisp**

	C	Lisp
<i>0 arguments</i>	<b>beep();</b>	<b>(beep)</b>
<i>1 argument</i>	<b>printf ("hello\n");</b>	<b>(message "hello")</b>
<i>2 arguments</i>	<b>max(10, 20);</b>	<b>(max 10 20)</b>
<i>3 arguments</i>	<b>max(10, 30, 20);</b>	<b>(max 10 30 20)</b>
	<b>3 + 4</b>	<b>(+ 3 4)</b>
	<b>3 + 4 + 5</b>	<b>(+ 3 4 5)</b>

In Lisp, everything is a function call. The function name and the arguments go inside the parentheses, and there are no commas or semicolons. Put a blank between words.

Experiment with the above Lisp expressions: go to the **\*scratch\*** buffer, type in a function call, and press the **LINEFEED** key (or **Ctrl-Enter** on an IBM PS/2 at LC-8 Tisch or the North Dorm).

**C-x C-b**                                 *Display a list of your buffers.*  
**C-x 1**                                    *Remove the list (the digit one).*  
**C-x b \*scratch\***                        *Go to the **\*scratch\*** buffer.*  
**(max 10 20) LINEFEED**

The arithmetic functions have non-alphabetic names. Space them exactly as shown.

**(+ 1 2) LINEFEED**  
**(- 1 2) LINEFEED**  
**(% 28 5) LINEFEED**                         *remainder*  
**(\* (+ 1 2) (+ 3 4)) LINEFEED**             *In C, (1+2) \* (3+4) == 21*

Functions that return **t** or **nil** (for “true” and “false”) are called *predicates*:

<code>(= 1 2) LINEFEED</code>	<i>equal; compare numbers or characters</i>
<code>(/= 1 2) LINEFEED</code>	<i>not equal</i>
<code>(&lt;= 1 2) LINEFEED</code>	<i>less than</i>
<code>(and (&lt;= 1 2) (&lt;= 3 4)) LINEFEED</code>	
<code>(or (&lt;= 1 2) (&lt;= 3 4)) LINEFEED</code>	
<code>(string= "Moe" "Larry") LINEFEED</code>	<i>equal; compare strings</i>
<code>(not (string= "Moe" "Larry")) LINEFEED</code>	

### Useful Lisp functions

The following functions operate on the text at the “point” immediately to the left of the cursor. To find the position of the point,

<code>(point)</code>	<i>number of characters from start of file</i>
<code>(current-column)</code>	<i>number of characters from start of line</i>
<code>(what-line)</code>	<i>displays number of lines from start of file; no return value</i>
<code>(bolp)</code>	<i>t if the point is at the start of the line</i>
<code>(eolp)</code>	<i>t if the point is at the end of the line</i>
<code>(bobp)</code>	<i>t if the point is at the start of the file</i>
<code>(eobp)</code>	<i>t if the point is at the end of the file</i>

To move the point,

<code>(forward-char 3)</code>	<i>can also take a negative number</i>
<code>(forward-line 3)</code>	<i>can also take a negative number</i>
<code>(beginning-of-line 1)</code>	<i>The number one means the current line.</i>
<code>(end-of-line 1)</code>	<i>The number one means the current line.</i>
<code>(goto-line 3)</code>	<i>line number, counting from start of the file</i>
<code>(goto-char 3)</code>	<i>character number, counting from the start of the file</i>
<code>(skip-chars-forward "A-Z")</code>	<i>the wildcard [A-Z] without the square brackets</i>
<code>(skip-chars-backward "^A-Za-z")</code>	

To examine the character on either side of the point,

<code>(following-char)</code>	<i>returns the character to the right of the point</i>
<code>(preceding-char)</code>	<i>returns the character to the left of the point</i>
<code>(looking-at "[Mm]oe\$")</code>	<i>t if text after point matches regular expression</i>

To insert and delete characters,

<code>(insert "hello")</code>	<i>insert a string</i>
<code>(insert ?A)</code>	<i>insert a single character; ?A means 'A' (p. 233)</i>
<code>(delete-char 3)</code>	<i>positive to delete on right, negative on left</i>
<code>(kill-line 1)</code>	<i>kill everything from point to end of current line</i>

### Numeric variables

<code>(setq i 10) LINEFEED</code>	<i>assignment statement</i>
<code>i LINEFEED</code>	<i>Value is now 10.</i>
<code>(quote i) LINEFEED</code>	<i>Value is i.</i>
<code>'i LINEFEED</code>	<i>Single quote, another way to do the same thing.</i>
<code>(+ i 5) LINEFEED</code>	<i>Should print 15.</i>
<code>(* i (+ i 5)) LINEFEED</code>	<i>Should print 150.</i>

**String variables**

```

(setq s "watermelon")           ;assignment statement
(length s)                       ;Value is 10.
(string-match "m[aeiou]" s)     ;Value is 5.

(substring s 2 5)                ;Value is "ter".
(substring s 5)                  ;Value is "melon".
(substring s -2)                 ;Value is "on".

(concat "water" "melon")        ;Value is "watermelon".
(setq s1 (concat "water" "melon")) ;assignment statement

```

**How to define your own Lisp function**

Write the following function definition in your `$HOME/.emacs` file. Function and variable names are usually made of lowercase letters and dashes.

```

1 (defun encomment ()
2   "Surround the current line with a pair of C comment delimiters."
3   (interactive "**")
4
5   (beginning-of-line 1)
6   (insert "/* ")
7   (end-of-line 1)
8   (insert " */")
9 )
10
11 ;Bind the above function to the C-t key.
12 (global-set-key "\C-t" 'encomment)

```

To execute this function, you must first execute your `$HOME/.emacs` file. Do this by exiting from `emacs` with `C-x C-c` and run `emacs` again. Edit a file with `C-x C-f`, move the cursor to a line you want to comment out, and execute your function `encomment` by typing

**M-x encomment RET**

Instead of executing a function by typing `M-x` and its name, you can bind the function to a key as shown above. Write the binding instruction after the end of the function definition.

**A more elaborate function**

```

1 (defun encomment ()
2   "Surround the current line (excluding leading blanks and tabs)
3   with a pair of C comment delimiters. Then return the point to
4   its original position."
5   (interactive "**")
6
7   ;Remember where we are now.
8   (setq i (point))
9
10  (beginning-of-line 1)
11  (skip-chars-forward " ") ;Skip leading blanks and tabs.
12  (insert "/* ")
13  (end-of-line 1)
14  (insert " */")
15

```

```

16      ;Go back to where we were at the beginning, allowing for the
17      ;extra three characters "/* ".
18      (goto-char (+ i 3))
19 )

```

Since the double-quoted string that you **insert** can contain a `\n`, you can easily write a function to insert the line

```
#if 0
```

before the current line, and insert the line

```
#endif
```

after the current line.

You can also write a function to surround the current line with a box. In other words, if the current line is

```
This is a comment.
```

the function will change it to

```

/*****/
/* This is a comment. */
/*****/

```

Measure the length of the current line to determine the number of asterisks you have to write. Then write a **while** loop.

```

(beginning-of-line 1)
(setq i (point))
(end-of-line 1)
(setq len (- (point) i))      ;length of the current line

```

### while loops

A **while** loop is really a function:

```
(while arg1 arg2 arg3 etc. )
```

The first argument should be a predicate such as

```
(<= i 10)
```

The other arguments constitute the body of the loop. For example,

```

(setq i 1)                ; i = 1;
(while (<= i 10)         ; while (i <= 10) {
  (message "%d" i)
  (sleep-for 1)
  (setq i (+ i 1))      ; ++i;
)

```

```
;Delete leading blanks and tabs from the start of the current line.
```

```

(beginning-of-line 1)
(while (looking-at " ")
  (delete-char 1)
)

```

Write a function that will delete leading and trailing blanks and tabs from the current line, and then add leading blanks to center the line. Assume the maximum line length is 80.

**This line is centered.**

### if statements

The **if** function can take two arguments:

```
(if arg1 arg2)
```

The first argument must have the value **t** or **nil**; use a predicate such as

```
(= (following-char) ?A)
```

The second argument is the “then” section: the function to be called if the first argument was **t**. Use a simple function such as

```
(message "The character after the point is an uppercase A.")
```

for your initial experiments. The finished **if** function looks like this:

```
(if (= (following-char) ?A)
    (message "The character after the point is an uppercase A.")
  )
```

If the “then” section consists of two or more function calls, you must surround it by (**progn** and **)**, which have the same meaning as { and } in C:

```
(if (= (following-char) ?A)
    (progn
      (message "The character after the point is an uppercase A.")
      (sleep-for 3)
    )
  )
```

The **if** function can also take any number of additional arguments:

```
(if arg1 arg2 arg3)
(if arg1 arg2 arg3 arg4)
```

These additional arguments constitute the “else” section. For example,

```
(if (= (following-char) ?A)
    (message "The character after the point is an uppercase A.")
    (message "The character after the point is not uppercase A.")
  )
```

Unlike the “then” section, the “else” section does not have to be surrounded by (**progn** and **)** when it consists of two or more function calls.

```
(if (= (following-char) ?A)
    (message "The character after the point is an uppercase A.")

    (message "The character after the point is not uppercase A.")
    (beep)
  )
```

```
(if (= (following-char) ?A)
  (progn
    (message "The character after the point is an uppercase A.")
    (sleep-for 3)
  )

  (message "The character after the point is not uppercase A.")
  (beep)
)
```

### if-then-else-if

As in C, the “else” section may consist of another `if`.

```
(if (= a b)
  (message "a equals b.")
  (if (< a b)
    (message "a is less than b.")
    (message "a is greater than b."))
))
```

### Complicated logical expressions

```
(if (= (preceding-char) (following-char))
  (message "I'm surrounded by two identical characters.")
)

;Three ways to write the same thing.
(if (and (= (preceding-char) (following-char)) (= (following-char) ?A))
  (message "I'm surrounded by two A's.")
)

(setq i (and (= (preceding-char) (following-char)) (= (following-char) ?A)))
(if i
  (message "I'm surrounded by two A's.")
)

(setq i (following-char))
(if (and (= (preceding-char) i) (= i ?A))
  (message "I'm surrounded by two A's.")
)
```

### Sample Lisp function with control structure

Before defining your own Lisp functions, look at the function definitions (`defun`'s) in the file `/usr/local/gnuemacs/lisp/simple.el` and the other files in that directory. The easiest ones to read are `back-to-indentation` and `open-line`.

```
1 (defun comma ()
2   "If the point is in a string of digits, insert commas into
3   the string every three digits. Otherwise, beep and do nothing."
4   (interactive ""))
5
6   (skip-chars-forward "0-9")
7   (forward-char -1))
```

```

8      (if (not (looking-at "[0-9]"))
9          ;then
10         (beep)
11
12         ;else
13         (forward-char -3)
14         (while (looking-at "[0-9][0-9][0-9][0-9]")
15             (forward-char 1)
16             (insert ?,)
17             (if (>= (current-column) 5)
18                 (forward-char -5)
19             )
20         )
21     )
22 )

```

### A Lisp function with arguments

Here is how to make a Lisp function that begins by asking you some questions. We've already seen this: for example, in the function `replace-regexp`.

```

1 (defun funky (n w)
2     "Sample function with two arguments. Write one documentation
3     string in double quotes after the defun line."
4     (interactive "\nPlease type a number: \nPlease type a word: ")
5
6     (message "You typed %d and %s." n w)
7     (sleep-for 3)          ;number of seconds
8     (beep)
9     (setq s1 (read-string "Please type another string: "))
10    (message "You typed %s." s1)
11 )

```

### Call a shellscript or C program from your Lisp function

To run another program, pass it command line arguments, and insert its output into the file you are editing,

```

(defun calendar ()
  "Insert a calendar."
  (interactive "")

  (insert ?\n)
  (call-process "cal" nil t nil "1" "2008")
)

```

The third argument is `t` to insert the program's output into the file.

```

(defun chmod+w ()
  "chmod +w the current file."
  (interactive "")

  (call-process "chmod" nil t nil "+w" buffer-file-name)
  (setq buffer-read-only nil)
  (message "chmod +w %s" buffer-file-name)
)

```



To run another program and feed it a region of the file being edited,

```
(defun lower ()
  "Change the next three characters to lowercase."
  (interactive "*")
  (call-process-region (point) (+ (point) 3) "tr" t t nil "A-Z" "a-z")
)
```

The first two arguments of `call-process-region` are the starting and ending position of the region you want to feed it. The fourth argument is `t` to delete the region from the file. The fifth argument is `t` to insert the programs input into the file.

### ▼ Homework 1.12: Define a useful Lisp function

Here are some ideas for a simple function. Define it in your `.emacs` file and bind it to a key.

- (1) If the point is in a number with a decimal point, round (or truncate) it to 5 (or  $n$ ) decimal places. Otherwise, beep.
- (2) If the point is in a word such as **Drive**, **Street**, **Boulevard**, etc., change the word to **Dr.**, **St.**, **Blvd.**, etc.; and vica versa. Otherwise, beep.
- (3) If the point is in a word that has different American and British spellings, toggle the spelling. Otherwise, beep.
- (4) Use the `message` function to display the length of the current line.
- (5) Put C comment delimiters `/* */` around the current line. Do not include leading tabs or spaces within the comment delimiters; see the definition of the function `back-to-indentation` in the file `/usr/local/gnuemacs/lisp/simple.el` for inspiration.
- (6) Do what lowercase `o` does in `vi`. Write a similar function for uppercase `O`; see the definition of the function `open-line` in the file `/usr/local/gnuemacs/lisp/simple.el` for inspiration.
- (7) Remove trailing blanks and tabs from every line in the file. Then return the cursor to the place where it was.
- (8) Implement a function with four arguments (two numbers and two strings) to perform the `vi` function

```
:10,20s/moe/larry/g
```

Use the `interactive` arguments shown above in the `funky` function.

□