Themes

- *Software is fun*

  be creative and clever

  acquire good grasp of fundamentals

- *Software is serious*

  people will depend on your code

  writing robust code is difficult

Becoming a good programmer

- *Study the programs of others*

- *Experiment*

- *Design & write as much code as possible*

Homework for Wednesday (2/7)

1. log into your `phoenix` account

2. send me e-mail with your web page, etc.

3. read man pages for the unix commands/programs:

   `man`

   `cc` (`gcc`)

   `make`

   `diff`

   `dbx` (`gdb`)

   for documentation on Gnu software (`gcc`, `gdb`), need to set `MANPATH` environment variable to point to `/usr/princeton/man`

4. read man pages for the system calls:

   `open`, `close`, `read`, `write`, `lseek`, `creat`

   `mkdir`, `rmdir`, `chdir`

   `getpid`, `getppid`, `fork`

   (What's the difference between a system call and a library routine, e.g. `open` vs. `fopen`?)

   Note: system calls are in section 3 of the man pages

```
$ ls
small_project/  large_project/
$ cd small_project/
$ ls
README          hello*          hello.d         hello.output
TODO            hello.c         hello.h         makefile
$ cat README
The "hello, world" program.

makefile -  builds "hello"
hello.c  -  the source
hello.h  -  defines the content of the "hello" string
$ cat hello.c
/* hello.c */

#include <stdio.h>
#include "hello.h"

void main(argc, argv)
int argc;
char *argv[];
{
  printf("%s!\n", HELLO_STRING);
}

/* end of hello.c */
$ cat hello.h
/* hello.h */

#define HELLO_STRING "Hello, world"

/* end of hello.h */
$ cat TODO
make program say hello in other (human) languages
test program on other computers and operating systems
$
```

```
$ cd large_project/
$ ls
README  TODO    bin/    src/
$ ls bin
run_me*        run_me.debug*
$ cd src
$ ls
README      driver.c    graphics.c  include/    main.c      makefile
$ ls include
macros.h  util.h
$
```