Sabrina Wong (861195047)
Apollo Truong (861168740)
Lab 1 Design Document

Please make sure that you will

1. Explain how you would add your own hello world program so that you are able to execute it from the xv6 shell

To add the program (hello world):
Similarly how we implemented lab1.c,
In the Makefile:

- We added “_helloworld” into /{UPROGS} (line#177)
- At EXTRA=/, we added (line#246) helloworld.c

We will then call

```
$ make clean; make
```

Then

```
$ make qemu-nox
```

Then

```
$ ./helloworld
```

2. Track the operation of the wait system call using the debugger. Explain what happens when the process has no children, and when it has a child process.

Wait will halt the operation that is currently running and look for the processes that are zombie children under the process. If the process has **no** children, the wait will finish the process currently running. If there **are** children, the process will wait for the children to finish first.

Files Changed:
defs.h
proc.h
proc.c
user.h
syscall.h
syscall.c
sysproc.c
sysproc.h
ln.c: exit(0);
zombie.asm: exit(0);
init.asm: exit(0);

```
forktest.c:  exit(0);
trap.c:    exit(0);
stressfs.c:  exit(0);
cat.asm:    exit(0);
rm.asm:    exit(0);
initcode.asm:# for(;;) exit(0);
mkfs.c:  exit(0);
initcode.S:# for(;;) exit(0);
grep.c:  exit(0);
kill.asm:    exit(0);
echo.c:  exit(0);
forktest.asm:    exit(0);
cat.c:  exit(0);
wc.asm:    exit(0);
usertests.asm:    exit(0);
usertests.asm:    exit(0);
grep.asm:    exit(0);
echo.asm:  exit(0);
ln.asm:    exit(0);
rm.c:  exit(0);
stressfs.asm:  exit(0);
sh.c:  exit(0);
zombie.c:  exit(0);
kernel.asm:      exit(0);
mkdir.asm:    exit(0);
mkdir.c:  exit(0);
ls.c:  exit(0);
kill.c:  exit(0);
init.c:      exit(0);
ls.asm:    exit(0);
usertests.c:  exit(0);
wc.c:  exit(0);
sh.asm:    exit(0);
```