Nicholas LaJoie, ECE 331, HW 6

```
// Author: Nicholas LaJoie
// ECE 331 - Homework 6
// Date: February 24, 2017
-------------------------
1. Making zombies (C code)
    a. Source Code
// Author: Nicholas LaJoie
// ECE 331 - Homework 6
// February 26, 2017
// Description: A C program that creates zombies by calling fork().
// Sources: http://askubuntu.com/questions/111422/how-to-find-zombie-process, http://stackover
flow.com/questions/6501522/how-to-kill-a-child-process-by-the-parent-process, http://man7.org/
linux/man-pages/man2/kill.2.html

#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <signal.h>

int main(int argc, char * argv[])
{

    pid_t child;

    // Create child process
    child = fork();

    // Create a Zombie
    if (child > 0) {
         // Parent sends a term signal to child, sleeps
        kill(child, SIGTERM);
        sleep(10);
    } else if (child == 0) {
        // Child sleeps in endless loop
        while (1) {
            sleep(1);
        }
    } else {
        perror("Failure on fork()\n");
        return 1;
    }
    return 0;
}

    b. Makefile
TARGET=zombie
CFLAGS=-g -Wall
OBJS=zombie.o
LIBS=
CC=gcc

.PHONY: all clean

all: ${TARGET}

${TARGET}: ${OBJS}
        ${CC} -o ${TARGET} ${OBJS} ${LIBS}

clean:
        rm -f ${TARGET} ${OBJS}

    c. Commands for proof:
```

Nicholas LaJoie, ECE 331, HW 6

To prove the creation of a zombie, I ran the zombie executable, and while it was running, entered "ps aux | grep 'Z\+'" in a different terminal, which displayed:
"pi 18497 0.0 0.0 0 0 pts/0 Z+ 13:32 0:00 [zombie] <defunct>"
Therefore, a zombie was created!

2. SD Card Geometry
a. Makefile

```
TARGET=SDgeo
CFLAGS=-g -Wall
OBJS=SDgeo.o
LIBS=
CC=gcc

.PHONY: all clean

all: ${TARGET}

${TARGET}: ${OBJS}
	${CC} -o ${TARGET} ${OBJS} ${LIBS}

clean:
	rm -f ${TARGET} ${OBJS}
```

b. Source Code

```
// Author: Nicholas LaJoie
// ECE 331 - Homework 6
// February 26, 2017
// Description: C program that prints the geometry of an SD cad using HDIO_GETGEO
// Sources: www.kernel.org/doc/Documentation/ioctl/hdio.txt

#include <stdio.h>
#include <linux/hdreg.h>
#include <sys/ioctl.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>

int main(int argc, char * argv[])
{
   struct hd_geometry geo;
   int fd, result;

   // Open SD card partition file
   fd = open("/dev/mmcblk0", O_RDONLY);
   if (fd < 0) {
       perror("Error getting file descriptor to SD card.\n");
       return 1;
   }

   // Get geometry of sd card
   result = ioctl(fd, HDIO_GETGEO, &geo);
   if (result == -1) {
       perror("Error calling HDIO_GETGEO.\n");
       return 2;
   }

   // Print geometry of sd card
   printf("Heads: %u\nSectors: %u\nCylinders: %hu\nStart: %lu\n", geo.heads, geo.sectors, geo.
cylinders, geo.start);

   return 0;
}
```

Nicholas LaJoie, ECE 331, HW 6

3. Disble login for RPi serial port: Completed - had to follow different tutorial for RPi3 (http://spellfoundry.com/2016/05/29/configuring-gpio-serial-port-raspbian-jessie-including-pi-3/)

4. Enable I2C on RPi: Completed

5. Determine processes: ps -e | sed ’1d’ | wc -l

6. Virtual memory used by processes: ps augx | tee save | sed ’1d’ | awk ’{print $5}’ | perl -ne ’chomp;s/$/+/;print;’ | sed -e ’s/$/0\n/’ | bc -l
NOTE: To verify this expression, "tee save" produces a file "save" that contains the processes information. I brought it into excel and verified the summation of the virtual memory (it worked!)

7. Find >10 MB files in /usr/share directory: find /usr/share/ -size +10M
NOTE: Tested this by place a 206MB file in the /usr/share/ directory - the file was found!