\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

NAME

ECS150, FQ2013

MT

Oct. 29 , 2013

This MT contains 8 questions, for all of which you should attempt to provide answers. We are eager to give you partial credit for partial answers, but only if your partial answers provide sufficient detail.

For questions, 1-6, that ask you to indicate what the execution of a program will display please keep in mind the following:

* Indicate the possible orders in which values will be displayed and explain why the particular values are returned.
* The values the program will return might be uncertain. Please indicate situations where this is the case and characterize what values the program **might** return.
* Be sure to provide an explanation for your answers.

The exam is open book, open notes (even if they are only on your laptop), but not open neighbor or open Internet. *For sure, you are* ***not*** *to compile or execute the programs*.

Write your answers on the exam sheet.

Problem 1: 10 points: Here is a program:

#include <stdio.h>

#include <unistd.h>

int main()

{

int id;

printf("here comes the date.\n");

if (id = fork() == 0) {

printf ("PID is %d and ID is %d\n", getpid (), id);

execl ("/bin/date", "date", 0);

}

printf ("that was the date.\n");

}

Problem (2) 12 points. Here is a program:

#include <unistd.h>

#include <stdio.h>

#include <sys/types.h>

int main(void) {

pid\_t childpid;

pid\_t mypid;

pid\_t myparentspid;

mypid = getpid();

myparentspid = getppid();

childpid = fork();

if (childpid == 0)

printf ("ID(myPID\_1): %ld, ID(myPID\_2): %ld and Parent PID: %ld\n", (long int) getpid(), (long int) getppid(), (long int) myparentspid);

else

printf ("ID(myPID\_1): %ld, ID(myPID\_2): %ld and Parent PID: %ld\n", (long int) getpid(), (long int) getppid(), (long int) myparentspid);

return (0);

}

Problem (3) 12 points. What will the following program return when executed? After termination (if it terminates) what is the contents of junk.txt? Is there any uncertainty in the values returned? Again, explain your answers.

#include <stdio.h>

#include <fcntl.h>

int main()

{

int fd1, fd2, fd3;

fd1 = open ("junk.txt", O\_RDWR | O\_TRUNC);

printf("fd1 = %d\n", fd1);

/\* other statements, perhaps many \*/

write(fd1, "let's ", 5);

fd2 = dup2 (fd1, 1);

printf (" fd2 = %d\n", fd2);

write(fd2, " get ", 4);

fd3 = dup2 (fd1, 0);

printf(" fd3 = %d\n", fd3);

write(0," going\n", 6);

dup2(3, 2);

write(2, " !\n", 2);

}

Problem (4) 12 points. What will the following program return? Explain.

#include <stdio.h>

int main()

{

alarm(2);

printf("still going\n");

while (1);

printf("should this line be executed?\n");

}

Problem (5) 12 points. What will the following program return? Explain your answers.

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <stdio.h>

#include <string.h>

char phrase[] = "stuff that in your pipe";

int main()

{

int fd [2], bytesread;

char message [100];

pipe (fd);

if (fork() == 0)

{

close(fd[0]);

write(fd[1],phrase, strlen(phrase)+1);

while (1);

}

else

{

close (fd[1]);

bytesread = read(fd[0], message, 100);

printf("read %d bytes: %s\n", bytesread, message);

// bytesread = read(fd[0], message, 100);

// printf("read %d bytes: %s\n", bytesread, message);

close (fd[0]);

}

}

What if the two commented-out lines of code in the parent are included? Does this make a difference? Explain.

Problem 6: 12 points. What will the following program return? As a reminder, be sure to indicate output values for which there is uncertainty. Explain!!!

#include <stdio.h>

int main()

{

int pid;

printf ("I am the original process with PID %d and PPID %d\n", getpid (), getppid ());

pid = fork();

if (pid != 0)

{

printf (" I am the parent process with PID %d and PPID %d\n", getpid () , getppid());

printf ("my child's PID is %d\n", pid);

}

else

{

sleep (100);

printf ("I am the child process with PID %d an PPID %d \n", getpid() , getppid());

}

printf ("PID %d terminates. \n", getpid());}

Problem 7: 10 points

Suppose that the following processes arrive for execution at the times indicated. Each process will run the listed amount of time. In answering the questions, use nonpreemptive scheduling and base all decisions on the information that you have at the time the decision must be made. Show all work in computing your solution.

Process Arrival Time Service Time

A 0.0 8

B 0.4 4

C 1.0 1

1. What is the average turnaround time for these processes with the FCFS algorithm
2. What is the average turnaround time for these processes with the SPN algorithm?
3. Repeat (b) but let the CPU remain idle for the first 1 unit and then assume SPN scheduling.

Problem 8: 10 ponts.The figure below shows a schematic of the file descriptor table after a program executes the following:

myfd = open (“home/ann/my.dat”, O\_RDONLY);

Note that in class what the figure shows as the “system file table”, I called the filp table (see Tanenbaum, pg. 563). The *system file table*, which is share by all processes in the system, has an entry for each active opened file (or pipe). Each system file table entry contains the rw pointer, a count of the number of file descriptor table entries pointing to it, and the access mode.

Here are the questions:

(a) What happens when the process whose file descriptor table is shown in the figure above executes:

close(myfd) ?

(b) Do you believe the two entries in the system file table are associated with two different processes one of which is a parent and the other this parent’s child?