Operating Systems

Assignment: 1

Date:1st March 2018

Instructions:

1. **Assignment: 1 is for *all* section students of Operating Systems Spring 2018.**
2. **Assignment must be attempted individually, acquire help from *books* and the world of internet considering Teachers and class fellows unaware of OS.**
3. **Create a word doc named "Rollno_Section_A1", describe answers in and attach programming files with the help of object, for object, find help from the link. Answer every question corresponding to its sequence number.**
4. **DEAD LINE: 5th March, 2018, 3:00 AM**

Scheduling:

1. **Write the C program for *any one* of the following scheduling algorithms. Time should be taken in float.**
   - **a.** First Come first served
   - **b.** Round Robin
   - **c.** Shortest Job First

Processes:

2. **Write a c program which will fork a child process.**
   - **a.** Parent process will wait for the child process to complete.
   - **b.** Child process will execute a shell script by passing 2 arguments.
   - **c.** Shell Script would perform addition, subtraction, multiplication and division of that numbers.

3. **The program will fork four child process, each terminating in a different way:**
   - **a.** The Wise Son: sleeps for 1 second and the exits
   - **b.** The Simple Son: calls exec to display mac address and exits
   - **c.** The Wicked Son: sleeps for seven seconds and executes date command after waking up.
   - **d.** The Son Who Doesn't Know How to Ask Questions: Prints the id of its grandparent.

Hint: In the parent display messages for each child when every child exits. (Hint1: code for wait and exit codes is attached below

```
 #include < sys/types.h >
 #include < stdio.h >
 #include < unistd.h >
 #define SIZE 5
 int nums[SIZE] = { 0,1,2,3,4 } ;
 int main()
 {
 int i;
 pid t pid;
 pid = fork();
 if (pid == 0) {
 for (i = 0; i < SIZE; i++) {
 nums[i] *= i;
 printf("CHILD: %d ",nums[i]); /* LINE X */
} }
 else if (pid > 0) {
 wait(NULL);
 for (i = 0; i < SIZE; i++)
 printf("PARENT: %d ",nums[i]); /* LINE Y */
 }}return 0; }
```

4. **The following program uses processes. You have to compile and execute the program and conclude the experiment describing process behavior.**

```
#include <sys/time.h>
#include <stdio.h>
#include <pthread.h>
#include <unistd.h>
#include <stdlib.h>
#include <errno.h>
void *proc();
int shared_number;
main()  {
  int i;
  pthread_t new_thread;
  int sleep_time;
  int pid;
  int status;
  int seed;
  shared_number = 1;
  printf("Enter a positive integer for seed: ");
  scanf("%d",&seed);
  srand48(seed);                /* initialize random number stream */
  if ((pid = fork()) == 0)
    {                               /* child process */
      proc();
      exit(0);
    }
  else if (pid == -1)                  /* error        */
    {
      printf("error %d\n",errno);
      exit(-1);
    }
  else { /* parent process */
          for (i = 0; i < 50; i++){
             printf("MAIN PROCESS: i = %d, shared_number = %d\n",i,shared_number);
             sleep_time = 100000.0*drand48(); /* generate random sleep time */
             printf("sleep time = %d microseconds\n",sleep_time);
             usleep(sleep_time);
             shared_number = shared_number + 2;
            }
      wait(&status);           /* wait for child process */
      printf("MAIN PROCESS: DONE\n"); }
}
void *proc() {
  int i;
  int DONE;
  DONE = 0;
  i = 0;
  while (!DONE) {
      i++;
      if (i%10000 == 0)
            printf("CHILD PROCESS: i = %d,shared_number = %d\n",i,shared_number);
      if (i > 5000000)
            DONE = 1;
    }
  printf("CHILD PROCESS: DONE\n");
  }
```

Shell scripting:

5. **Write shell script for questions given below.**

   a. Create a directory named “programs” in your home directory. In this directory add hundred (100) files with extension of “.tiff”, ‘.c’, ‘.sh’. The size of each file will be between 2kb and 100kb. “truncate is used for size allocation” e.g. **truncate –s 2k file1.c**

   b. Change the extension of all “.jpg” files to ‘.jpeg’.

   c. Display all ‘.sh’ files with size more than 5kb.

   d. Create another directory Imp_programs” in the home directory and move all .jpg and/or .jpeg files from the ‘myprograms’ directory to this directory whose size is larger than 10kb.

   e. Display files in the current directory with creation time.

f. Rename all files in directory ‘myprograms’ with extension ‘.jpg’ in such a way that ‘img_’ is appended to the beginning their filename. e.g ‘f1 .jpg’ will be renamed as ‘img_f1.jpg’

g. Create a subdirectory ‘myexecutables’ in the directory ‘myprograms’. Move all executable files into this subdirectory from ‘myprograms’ directory.

h. Move files into three subdirectories directories: shelldir, cdir, jpgdir according to their extensions.

Hint: Use ls, grep, mkdir, mv, copy, for loop, if, else, truncate, dd switch for conditions. In case of any ambiguity, first check the command manual “man truncate, man dd etc”.

“Smart people learn from everything and everyone, average people from their experience and stupid people already have all the answers.” Socrates

Give your Best