Departamento de Ciencias e Ingeniería de la Computación

Ingeniería en Computación
Ingeniería en Sistemas de Información

Sistemas Operativos

Segundo Cuatrimestre de 2021

Actividades del Segundo Laboratorio

- Realizar todos las actividades planteadas.
- Entregar un informe describiendo las herramientas que utilizaron, los conceptos aprendidos y cómo se relacionan con lo visto en la teoría. El informe contendrá las respuestas a todos los incisos y conclusiones generales. Además deben incluir en la entrega los archivos fuentes que resuelven los ejercicios 1, 3 y 5.
- La entrega de la actividad se realizará a través del aula virtual con el nombre de Actividad2. En el contenido del informe deben incluir los nombres de los integrantes de la Comisión.

1. Dado el siguiente código:

```
#include <signal.h>
#include <stdlib.h>
#include <sys/types.h>
#include <unistd.h>
#include <stdio.h>
#include <sys/wait.h>
int pid;
int main()
{
    int status, s_wait;
    int trata_alarma();
    pid = fork();
    if (pid!=0)
    {
        printf(".... \n");
        signal(SIGALRM, (void *)(int)trata_alarma);
        alarm(10);
        wait(&status);
        alarm(0);
    }
    else
    {
        printf(".... \n");
        while(1){};
        exit(1);
    }
}
int trata_alarma()
{
    kill (pid, SIGKILL);
}
```

a) Explique el comportamiento del código presentado.

b) ¿Es posible que se conozca cuál es el padre y cuál es el hijo? Justifique su respuesta.

c) Realice las modificaciones necesarias para que el padre cree dos procesos hijos que realizan el mismo comportamiento. Escriba esta modificación en un archivo fuente denominado Ejercicio1.c

2. Enumere y explique brevemente 4 system calls que estén asociados con manejo de procesos para UNIX/Linux.

3. Realizar el siguiente experimento: Considere que el proceso P_1 tiene que realizar un conjunto de tareas y para las mismas utiliza procesos que colaboren con él. El proceso P_1 tiene como objetivo sumar dos matrices de 5x5. La tarea la divide en tres partes:

   a) Lee el contenido de las dos matrices.

   b) Para realizar la suma utiliza procesos colaboradores (P_2,P_3,P_4, P_5 y P_6), cada uno realiza la suma de una fila y el resultado los guarda en un archivo denominado SumaFila(i).txt, donde i representa al número de fila. Los procesos colaboradores trabajan concurrentemente (en paralelo)

   c) Cuando todos los procesos colaboradores terminan une el resultado y lo muestra por pantalla.

4. Dado el siguiente código en lenguaje C:

```
int a=0; //es una variable global

printf("Comienza a ejecutarse el proceso %d \n", getpid());
for (i=0; i < NUMHILOS; i++) {
        printf("Se crea un nuevo thread \n");
        pthread_create(&hilos[i], NULL, funcion, (void *)i);
    }
for (i=0; i < NUMHILOS; i++) {
        printf("Finaliza el thread ... \n");
        pthread_join(hilos[i], NULL);
    }
...
}
void *funcion (void *data) {
     unsigned long int pthid;
     int j;
     pthid = pthread_self();
     j = 0;
     while (j < 10000) {
          a++;
          printf("Soy el thread %d con identificacion %lu correspondiente al proceso %d \n", data,
          pthid, getpid());
          j++;
     }
}
```

   a) Codifique y ejecute el código dado. Considere que el valor de NUMHILOS es 7.

   b) ¿Qué es lo que realiza el código?¿Se puede identificar en qué orden terminan?

   c) ¿Cuál es el valor final de a?

5. Resolver el mismo problema que el inciso 3, utilizando threads para su solución.