

APRIL 12, 2021

**GTU Department of Computer Engineering
CSE344 - Spring 2021
Homework 2 Report**

**Akif Kartal
171044098**

1 Problem Definition

The problem is to estimate these 8 functions through polynomial interpolation using the Lagrange form in two rounds with signals.

2 Solution

2.1 Some Problems

2.1.1 Same Signal Multiple Times

In the 1st round each child will let the process m know (via a signal) that it has finished its calculations. But the problem was here, because of same signal type sending multiple times, kernel convert it into one signal therefore we can't recognize last child finished its job.

2.1.2 Solution

To solve this problem I used **sigsuspend** as recommended in this Homework. After 8 child created parent will signal them one by one such that you can start to work now.

2.1.3 Validation Coordinate

In homework document since in 1st round there will be using 6 coordinate I assumed 7th coordinate as for validation for fist round and I solved the problem according to this. In second round 8th coordinate was used for validation.

2.2 Error Handling

In order to handle any error **stderr** used with **write system call** and **exit** system call was used.

► **Note:** In this part **fprintf** library function was not used. Instead write system call was used.

Following code shows an example of this;

```
1 void myStderr(const char *str)
2 {
3     ssize_t size = strlen(str);
4     if (size != write(STDERR_FILENO, str, size))
5     {
6         perror("write system call error! ");
7         exit(EXIT_FAILURE);
8     }
9 }
```

Usage:

```
1 int rd = read(fd, buf, size);
2 if (rd == -1)
3 {
4     myStderr("reading error!\n");
5     exit(EXIT_FAILURE);
6 }
7 return rd;
```

2.3 Synchronization between Process

In 1st round synchronization provided mostly by using **sigsuspend** and **file locks**. In 2nd round synchronization provided by using **waitpid** and **sigsuspend** system calls and **file locks**.

2.4 CTRL-C Handling

In order to give a message on CTRL-C interrupt, I used **sigaction** function from **signal.h** library. Also, I used a **global variable** to set if an interrupt has occur.

```
1 volatile __sig_atomic_t exitSignal = 0;
2 void handler2(int signum){
3     if (signum == SIGINT)
4     {
5         //In case of CTRL-C
6         exitSignal = 1;
7     }
8
9 }
```

In each loop in code, signal flag was checked, on interrupt signal, resources was given back and exited elegantly.

2.5 Finding Coefficients

To find 7 coefficients of each polynomial's, I implemented following formula;

$$c_i = (-1)^{n-i} \frac{\det \begin{pmatrix} b_0 & b_1 & \dots & b_n \\ a_0^n & a_1^n & \dots & a_n^n \\ \dots & \dots & \dots & \dots \\ a_0^{i-1} & a_1^{i-1} & \dots & a_n^{i-1} \\ a_0^{i+1} & a_1^{i+1} & \dots & a_n^{i+1} \\ \dots & \dots & \dots & \dots \\ 1 & 1 & \dots & 1 \end{pmatrix}}{\det \begin{pmatrix} a_0^n & a_1^n & \dots & a_n^n \\ a_0^{n-1} & a_1^{n-1} & \dots & a_n^{n-1} \\ \dots & \dots & \dots & \dots \\ 1 & 1 & \dots & 1 \end{pmatrix}},$$

Figure 1

3 Test Result

A simple test result with given example file is following;

```
akif@ubuntu:~/Desktop/system-programming/HW2$ ./processM /home/akif/Desktop/source/example
Error of polynomial of degree 5: 8.8
Polynomial 0: -491.0,917.6,-547.1,153.0,-22.2,1.6,-0.0
Polynomial 5: -51.7,89.3,-48.3,12.1,-1.5,0.1,-0.0
Polynomial 2: 16.6,-8.4,-1.7,2.0,-0.4,0.0,-0.0
Polynomial 6: -1836.3,1791.2,-688.6,134.7,-14.2,0.8,-0.0
Polynomial 4: -109.8,211.1,-134.7,40.1,-6.0,0.4,-0.0
Polynomial 1: -13.0,37.5,-26.5,9.9,-2.0,0.2,-0.0
Polynomial 3: 799.0,-1125.8,615.2,-167.8,24.3,-1.8,0.1
Polynomial 7: 70.6,-113.3,69.4,-20.6,3.2,-0.3,0.0
Error of polynomial of degree 6: 9.6
akif@ubuntu:~/Desktop/system-programming/HW2$ make clean
```

Figure 2

4 References that was used

While doing this homework following references was used;

- ▶ Course Textbook Listing 26-5: Reaping dead children via a handler for SIGCHLD
- ▶ Lagrange's Interpolation
- ▶ Determinant Operation

Note:Homework was finished as expected. Only problem in 1st round to make error calculation 7th coordinate was used **not last one**.