**EXEC FUNCTION**

**Lab no# 04**

****

**Fall 2021**

**CSE-302 System Programming Lab**

Submitted by: **Ashfaq Ahmad**

Registration No: **19PWCSE1795**

Class Section: **B**

“On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work.”

Student Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Submitted to:

**Engr: Ma’am Madeha sheer**

**December** 5, 2021

**Department of Computer Systems Engineering**

**University of Engineering and Technology, Peshawar**

**Task 1:**

Write a program that takes N UNIX commands as arguments, creates N child processes, each of them implementing their respective commands. Parent process shall wait for all the child processes and receive and print the exit status of the child processes.

**Source Code:**

#include<stdio.h>

#include<unistd.h>

#include<stdlib.h>

#include<sys/wait.h>

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

{

int x;

for (int i=1; i<argc; i++)

{

x=fork(); //process fan will be created.No of child process will be according to no of commands.

if(x==0)

{

printf("PID: %d, Executing Command: %s\n",getpid(),argv[i]);

execlp(argv[i],argv[i],NULL);

break;

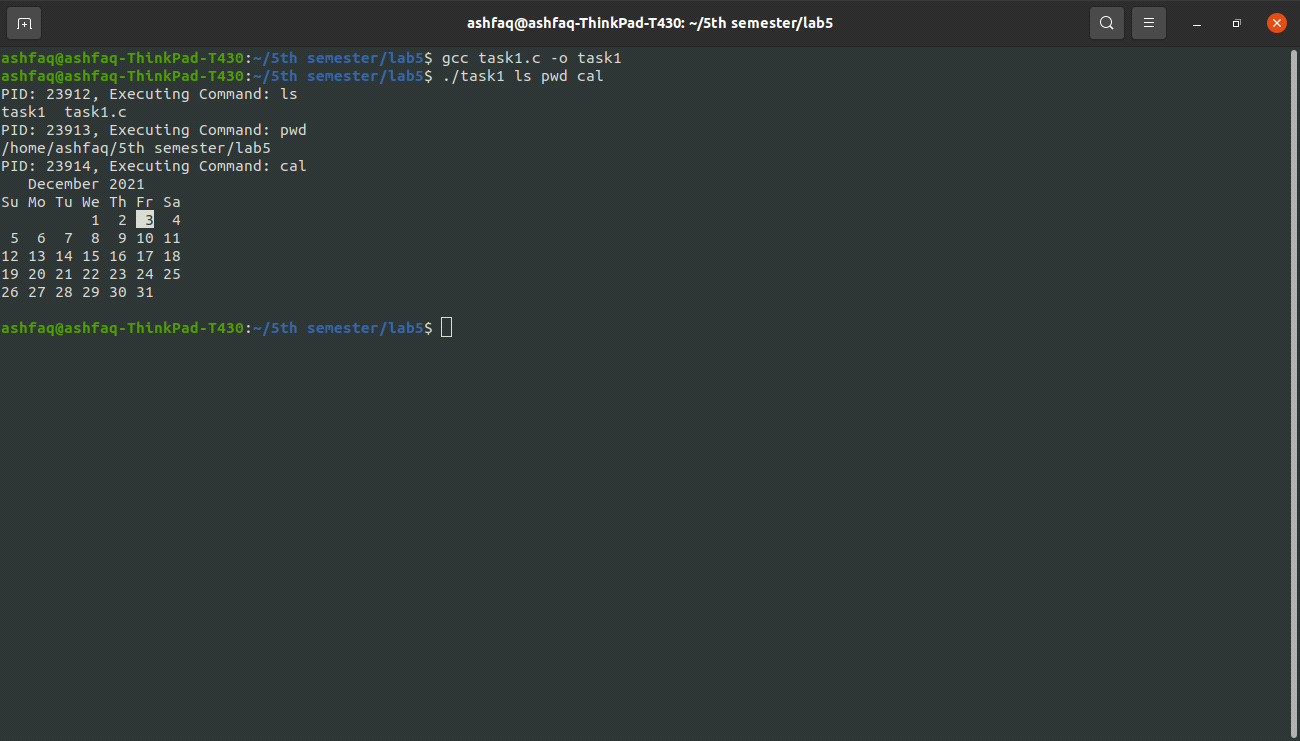
}

wait(1);

}

};

**Output:**



**Task 2:**

1. Write a program that takes integers as arguments and adds them.

**Source Code:**

#include<stdio.h>

#include<unistd.h>

#include<stdlib.h>

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

{

int sum=0;

for(int i=1;i<argc; i++)

{

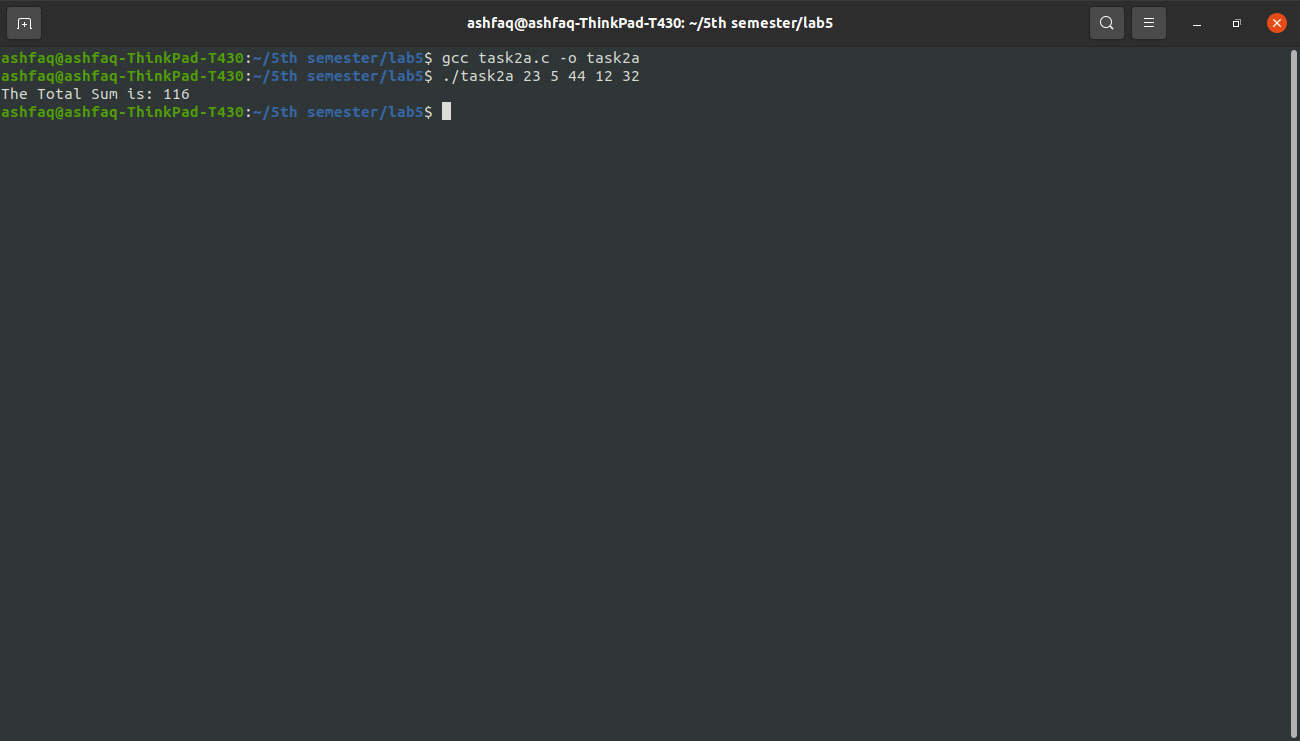
sum+=atoi(argv[i]);

}

printf("The Total Sum is: %d\n",sum);

}

**Output:**



1. Write a program that takes integers as arguments and multiplies them.

**Source Code:**

#include<stdio.h>

#include<unistd.h>

#include<stdlib.h>

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

{

int product=1;

for(int i=1;i<argc; i++)

{

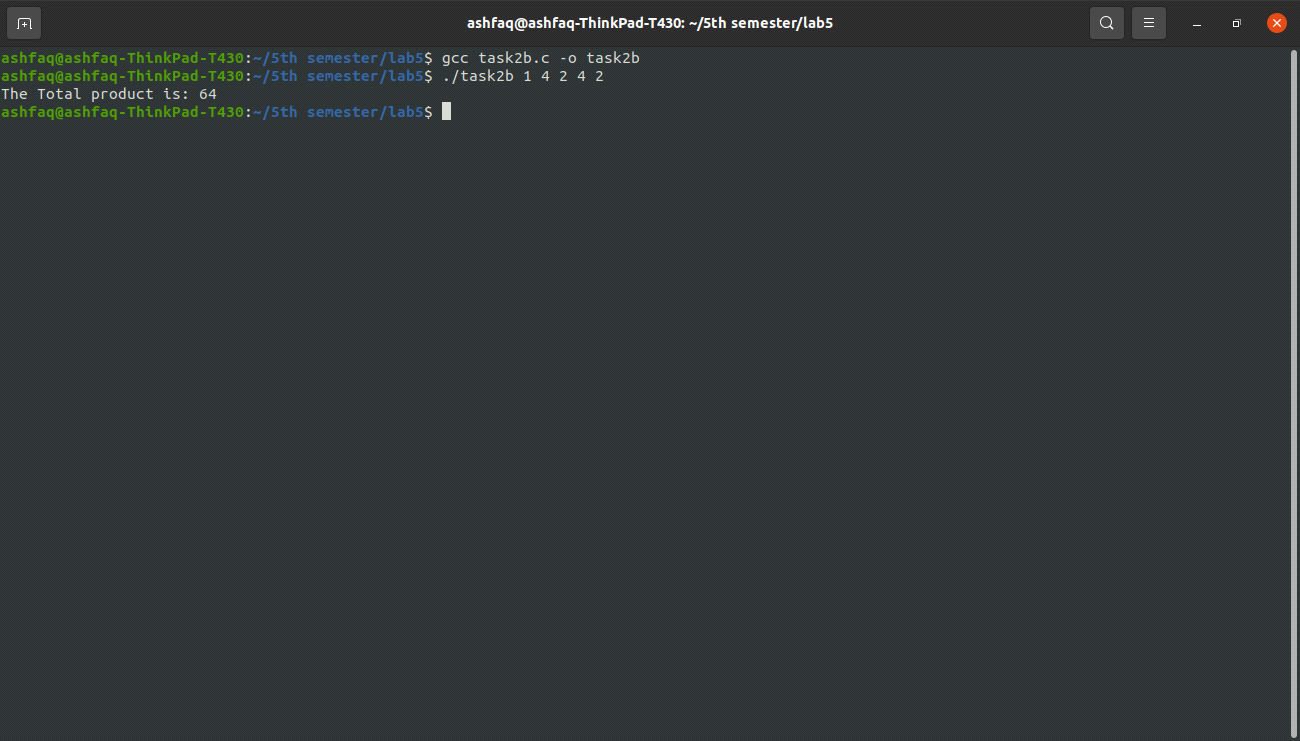
product\*=atoi(argv[i]);

}

printf("The Total product is: %d\n",product);

}

**Output:**



1. Write a program that takes integers as arguments & adds & multiplies them using the above two programs.

**Source Code:**

#include<stdio.h>

#include<unistd.h>

#include<stdlib.h>

#include<string.h>

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

{

for(int i=1; i<3; i++)

{

int x=fork(); //two child will create one for task2a execution 2nd for task2b execution

if(x==0)

{

if(i==1)

{

execvp("./task2a",argv);

}

else if(i==2)

{

execvp("./task2b",argv);

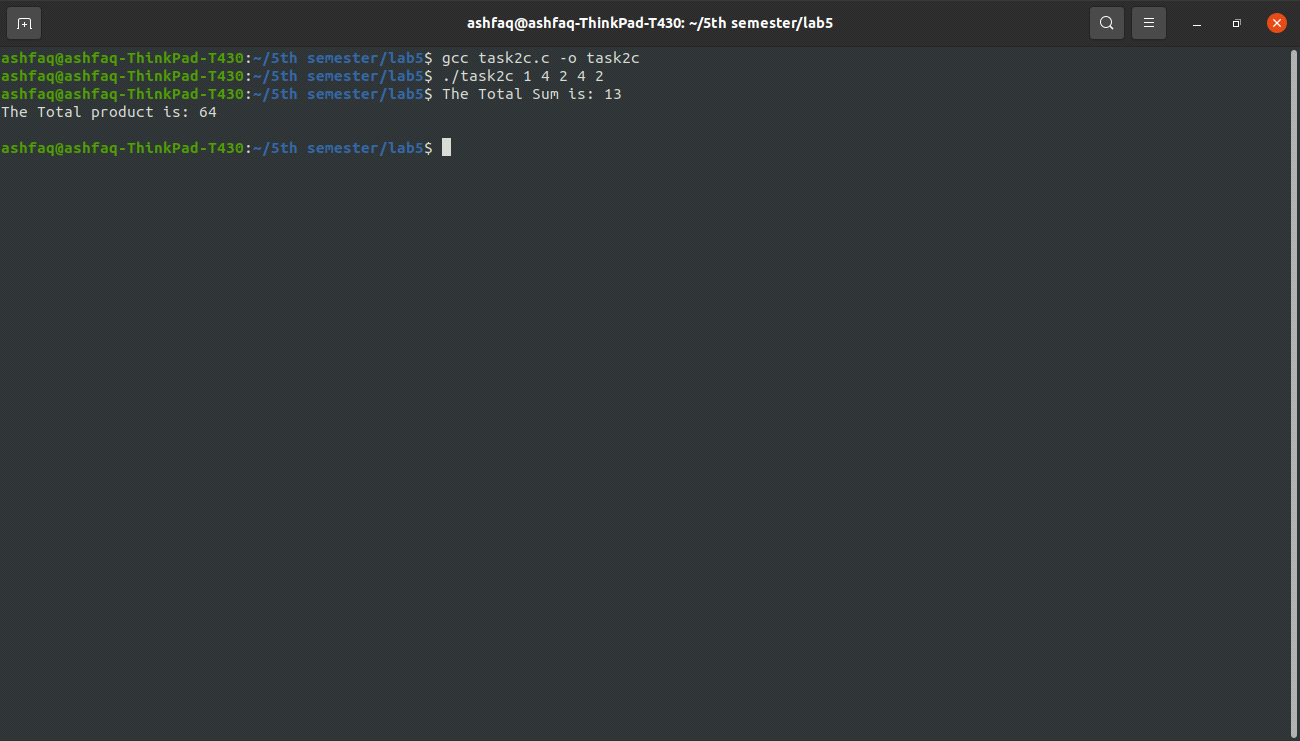
}

}

}

}

**Output:**



**Task 3:**

Write a program “minmax.c” that takes an array as command line arguments. Program executes min.c and max.c programs in its two child processes. One child processes calculates and returns the min value and other calculates and returns the max value in the array. The program “minmax.c” shall receive the values returned by the child processes and display these values.

**Source Code:**

**Max:**

#include<stdio.h>

#include<stdlib.h>

#include<unistd.h>

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

{

int max=atoi(argv[1]);

for(int i=1; i<argc; i++)

{

if(atoi(argv[i])>max)

{

max=atoi(argv[i]);

}

}

return max;

}

**Min:**

#include<stdio.h>

#include<stdlib.h>

#include<unistd.h>

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

{

int min=atoi(argv[1]);

for(int i=1; i<argc; i++)

{

if(atoi(argv[i])<min)

{

min=atoi(argv[i]);

}

}

return min;

}

**MaxMin:**

#include <stdio.h>

#include <unistd.h>

#include <stdlib.h>

#include <sys/wait.h>

#include <string.h>

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

{

int x,value;

for(int i=0; i<2; i++)

{

x=fork();

if(x==0)

{

if(i==0) //first child will execute this block

{

printf("Max:\n");

execvp("./max",argv);

perror("error status\n");

}

else //2nd child will execute this block

{

printf("Min:\n");

execvp("./min",argv);

perror("error status\n");

}

}

}

for(int i=0; i<2; i++)

{

int pid=wait(&value);

printf("PID: %d, status: %d\n",pid,WEXITSTATUS(value));

}

}

**Output:**



THE END