Name : Sarthak Pagar

Roll No. : 34

Classs : TE (IT)

Practical : 2A

Statement : [Implement the C program in which main program accepts the integers to be sorted. Main program uses the FORK system call to create a new process called a child process. Parent process sorts the integers using sorting algorithm and waits for child process using WAIT system call to sort the integers using any sorting algorithm. Also demonstrate zombie and orphan states.](https://drive.google.com/drive/folders/1tlK9yy11BahyceuY-tHNjttxFR4dWQz5?usp=sharing)

#include<stdio.h>

#include<unistd.h>

#include<sys/wait.h>

#include<stdlib.h>

#define MAX 20

void bubble\_asc(int arr[],int n)

{

int i,j,temp;

for(i=0;i<n;i++)

{

for(j=0;j<n-i-1;j++)

{

if(arr[j]>arr[j+1])

{

temp=arr[j];

arr[j]=arr[j+1];

arr[j+1]=temp;

}

}

}

}

void bubble\_dsc(int arr[],int n)

{

int i,j,temp;

for(i=0;i<n;i++)

{

for(j=0;j<n-i-1;j++)

{

if(arr[j]<arr[j+1])

{

temp=arr[j];

arr[j]=arr[j+1];

arr[j+1]=temp;

}

}

}

}

void print(int arr[],int n)

{

int i;

for(i=0;i<n;i++)

{

printf("\t%d",arr[i]);

}

printf("\n\n");

}

int main()

{

int i,n,arr[MAX],f,pid;

printf("\nHOW MANY NOS DO YOU WANT IN ARRAY : ");

scanf("%d",&n);

printf("ENTER ARRAY ELEMENT : ");

for(i=0;i<n;i++)

{

scanf("%d",&arr[i]);

}

pid=fork();

if(pid==0)

{

/\*ORPHAN STATE \*/

printf("\n\t\t\*\*\*\*\*\*\*\*\*\*ORPHAN STATE\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

sleep(5);

printf("\n\t CHILD PROCESS PID : %d ", getpid());

printf("\n\t PARENT PROCESS PPID : %d",getppid());

system("ps -el | grep init");

printf("\n\nARRAY IN ASSCENDING ORDER : ");

bubble\_asc(arr,n);

print(arr,n);

}

else if(pid >0)

{

/\*ZOMBIE STATE \*/

printf("\n\t\t\t\*\*\*\*\*\*\*\*\*\*ZOMBIE STATE\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

system("ps -el | grep Z");

f=wait(0);

printf("\n\tTERMINATED CHILD PID : %d",f);

printf("\n\tPARENT PID (MAIN) : %d",getpid());

printf("\n\tPARENTS PARENT PPID (BASH) : %d ",getppid());

printf("\n\nARRAY IN DESCENDING ORDER : ");

bubble\_dsc(arr,n);

print(arr,n);

}

else

printf("\nERROR IN FORK!!");

return 0;

}

Output :-

[Saru1594@localhost 2]$ gcc 2A.c

[Saru1594@localhost 2]$ ./a.out

HOW MANY NOS DO YOU WANT IN ARRAY : 5

ENTER ARRAY ELEMENT : 8

3

5

2

7

\*\*\*\*\*\*\*\*\*\*ZOMBIE STATE\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*ORPHAN STATE\*\*\*\*\*\*\*\*\*\*\*\*\*\*

F S UID PID PPID C PRI NI ADDR SZ WCHAN TTY TIME CMD

CHILD PROCESS PID : 12314

PARENT PROCESS PPID : 12307

ARRAY IN ASSCENDING ORDER : 2 3 5 7 8

TERMINATED CHILD PID : 12314

PARENT PID (MAIN) : 12307

PARENTS PARENT PPID (BASH) : 12248

ARRAY IN DESCENDING ORDER : 8 7 5 3 2