Date 16/10/19

Expt. No. DAY 8 Page No. 28

SIGNALS

2.3) WAP to print the default message of SIGINT and also print the user and a program for a process which cannot be killed by pressing ctrl + C and again restore the default status of it

Ans)

```
#include <stdio.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <unistd.h>
#include <signal.h>
void func (int signum)
{
    signal (SIGINT, SIG_DFL);
    printf ("\nsignal SIGINT received = %d, trying to stop program\n", signum);
}

void main()
{
    signal (SIGINT, func);
    printf ("process (%d) is running\n", getpid());
    sleep (2)
    while (1);
    {}
}
```

Teacher's Signature : _____

OUTPUT

```
[2yrgr2] $ gcc signal23.c -o toy
[2yrgr2] $ ./toy
        process (8934) is running
        ^C
        signal SIGINT received=2, trying to stop the program
        ^C
[2yrgr2] $
```

Date .....

Expt. No. .....

Page No. 24

PIPES

① WAP to implement how a process communicates with another process using the pipe function.

Ans)

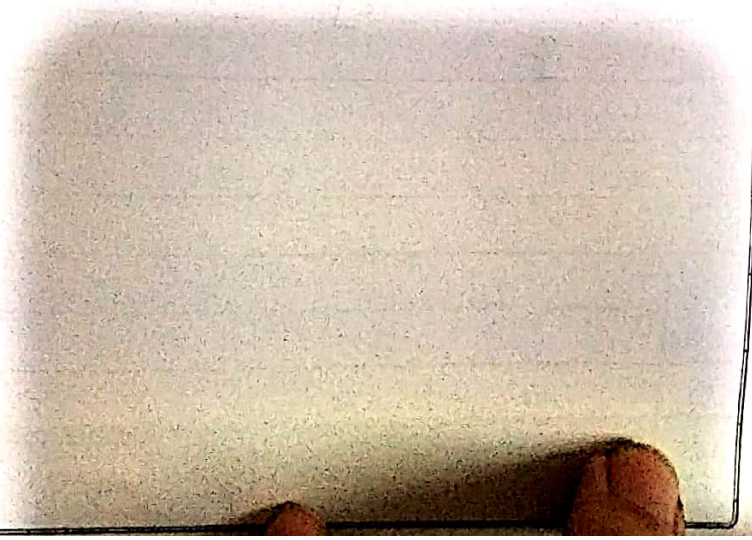
```
#include <stdio.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <unistd.h>
#include <string.h>
#define msgsize 100
char *msg = "hello os lab\n";
void main() {
    char inbuf[100]; pid_t childpid; int fd[2], i, bytes;
    pipe(fd); // pipe system call
    if ((childpid = fork()) == 0) {
        close(fd[0]); // close read end b4 write
        printf("child (%d) created and preparing to write message = %s
            with length = %d\n", getpid(), msg, strlen(msg));
        write(fd[1], msg, strlen(msg)+1);
        sleep(2); exit(0);
    }
    else if (childpid > 0) {
        close(fd[1]); // close write end b4 read
        bytes = read(fd[0], inbuf, sizeof(inbuf));
        printf("parent (%d) of child (%d) retrieves %d bytes
            of strings = %s\n", getpid(), childpid, bytes, inbuf);
        wait(NULL);
    }
}
```

Teacher's Signature : _____

OUTPUT

child (3750) created and preparing to write
message = hello os lab with length = 13

Parent (3751) of child (3750) retrieves 14 bytes of string =
hello os lab



(3) WAP that creates a one-way pipe between a parent and child process. The parent process gets a string from standard Input and sends the string to child. The child prints the reverse of the string. Both parent and child terminates when the string "quit" is the Input.

Ans)
```
#include <stdio.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <unistd.h>
int flag=0; pid_t childpid;
void main(){
    int fd[2], bytes, i, j, strsize; char str[40], readbuffer[80], rev[80];
    pipe(fd);
if( (childpid=fork())==0) {
        close(fd[1]);
        do{
            bytes=read(fd[0], readbuffer, sizeof(readbuffer));
            printf("child(%d) reads %d bytes of strings %s\n",
                getpid(), bytes, readbuffer);
            if(strcmp(readbuffer, "quit")==0)
                exit(0);
            else
                flag=1;
            i= strlen(readbuffer)-1; j=0;
            while(i>=0){
                rev[j]=readbuffer[i--];
                printf("%c", rev[j++]);
            } printf("\n");
        }while(flag);
}//end if
```

Teacher's Signature : _____

```
else if (childpid > 0)
{
    close(fd[0]);
    do
    {
        printf("parent (%d) enter string or quit to
                exit\n", getpid());
        scanf("%s", str);
        printf("string = %s entered with size %d\n",
                str, strlen(str));
        strsize = write(fd[1], str, strlen(str)+1);
        printf("Parent (%d) sent %d bytes of string - %s
        to with child (%d)\n", getpid(), strsize, str,
                childpid);
        sleep(2);
    } while(strcmp(str,"quit") != 0);
} //else if
} //main
```

☆☆☆☆☆

OUTPUT

```
Parent (9488) enter string or quit to exit
csen3113
String = csen3113 entered with size = 8
Parent (9488) sent 9 bytes of string = csen3113 to child (9489)
Child (9489) reads 9 bytes of string = csen3113
3113nesc

Parent (9488) enter string or quit to exit
yash
string = yash entered with size = 4
Parent (9488) sent 5 bytes of string = yash to child (9489)
Child (9489) reads 5 bytes of string = yash
hsay

Parent (9488) enter string or quit to exit
quit
string = quit entered with size = 4
Parent (9488) sent 5 bytes of string = quit to child (9489)
Child (9489) reads 5 bytes of string = quit
```