**LAST ASSIGNEMENT**

**Name:Abdullah talat**

**Reg no=18-arid-2595**

Q1

#include<stdio.h>

#include<unistd.h>

#include<pthread.h>

#include<sys/wait.h>

void \* thread\_fun(void \* arg){

printf("child process \n");

}

void \* thread1\_fun(void \* arg){

//printf("roll no=18-arid-2595\n");

pthread\_t p1;

pthread\_create(&p1,NULL,thread\_fun,NULL);

pthread\_join(p1,NULL);

printf("roll no=18-arid-2595\n");

}

void \* thread2\_fun(void \* arg){

pthread\_t p1;

//printf("name=abdullah talat\n");

pthread\_create(&p1,NULL,thread\_fun,NULL);

pthread\_join(p1,NULL);

printf("name=Abdullah talat\n");

}

int main(){

pthread\_t p1,p2;

printf("start process\n");

if(fork()==0){

pthread\_create(&p1,NULL,thread1\_fun,NULL);

pthread\_join(p1,NULL);

}

else{

wait(NULL);

pthread\_create(&p2,NULL,thread2\_fun,NULL);

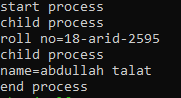
pthread\_join(p2,NULL);

printf("end process\n");

}

}

**output**



**Q2**

#include<stdio.h>

#include<unistd.h>

#include<stdlib.h>

int main(){

int pdf[2];

int g=pipe(pdf);

if(g==-1){

printf("piping dont occur");

exit(1);

}

int d=fork();

if(d==-1){

printf("child process dont create");

exit(1);

}

if(d==0){

dup2(pdf[0],0);

close(pdf[0]);

execlp("ls","ls",NULL);

}

else{

dup2(pdf[1],1);

close(pdf[1]);

execlp("ws","ws","-l",NULL);

}

}

**Output**

****