//Implement the C program for Reader-Writer problem with reader priority.

#include<stdio.h>

#include<stdlib.h>

#include<semaphore.h>

#include<pthread.h>

#include<unistd.h>

#include<stdint.h>

#define R 5

#define W 5

int readCount;

pthread\_mutex\_t x;

sem\_t wsem;

int s=5;

void \*reader1(void \*a);

void \*writer1(void \*a);

int main()

{

pthread\_mutex\_init(&x,NULL);

sem\_init(&wsem,0,1);

pthread\_t thread\_read[R],thread\_write[W];

int op;

do

{

printf("Enter the Choice:\n");

printf("Menu:1.Reader having priority 2.Exit:\n");

scanf("%d",&op);

switch(op)

{

case 1:readCount=0;

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

{

pthread\_create(&thread\_write[i],NULL,\*writer1,(void \*)(intptr\_t)i);

}

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

{

pthread\_create(&thread\_read[i],NULL,\*reader1,(void \*)(intptr\_t)i);

}

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

{

pthread\_join(thread\_write[i],NULL);

}

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

{

pthread\_join(thread\_read[i],NULL);

}

break;

case 2:break;

}

}while(op!=2);

return 0;

}

void \*reader1(void \*a)

{

int r=(intptr\_t)a;

int i=0;

while(i<5)

{

pthread\_mutex\_lock(&x);

readCount++;

if(readCount==1)

{

sem\_wait(&wsem);

}

pthread\_mutex\_unlock(&x);

printf("\t\tReader %d read %d.\n",r,s);

pthread\_mutex\_lock(&x);

readCount--;

if(readCount==0)

{

sem\_post(&wsem);

}

pthread\_mutex\_unlock(&x);

sleep(rand() %10);

i++;

}

}

void \*writer1(void \*a)

{

int w=(intptr\_t)a;

int i=0;

while(i<2)

{

sem\_wait(&wsem);

s+=5;

printf("Writer %d write %d.\n",w,s);

sem\_post(&wsem);

sleep(rand() %10);

i++;

}

}