/\*

Queues are frequently used in computer programming, and a typical example is the creation of a job queue by an operating system.

If the operating system does not use priorities, then the jobs are processed in the order they enter the system.

Write C++ program for simulating job queue. Write functions to add job and delete job from queue.

\*/

#include <iostream>

using namespace std;

#define size 5

class spq

{

int f,r,job,djob; //data members

int simpq[size],prioq[size];

public:

spq() //Default constructor

{

f=r=-1; //init front and rear to -1.

job=djob=0;

prioq[-1]=0;

}

//To check Q is full or not

int isQfull()

{

if(r==size-1)

return 1;

else

return 0;

}

//To check Q is empty or not

int isQempty()

{

if((f==-1)||(f>r))

return 1;

else

return 0;

}

void simpqadd(); //member functions.

void showsimpleQ();

void delsimpleQ();

void prioqadd();

void delprioQ();

void showprioQ();

};

//To insert the job inside the simple queue.

void spq::simpqadd()

{

cout<<"\nEnter the Job: ";

cin>>job;

if(isQfull())

cout<<"\nSorry !! Queue is full....\n";

else

{

if(f==-1)

{

f=r=0;

simpq[r]=job;

}

else

{

r=r+1;

simpq[r]=job;

}

}

}

//To delete job from the simple queue.

void spq::delsimpleQ()

{

if(isQempty())

cout<<"\nSorry Q is empty...\n";

else

{

djob=simpq[f];

f=f+1;

cout<<"\nDeleted job is: "<<djob;

}

}

//To show all the jobs from SimpleQueue.

void spq::showsimpleQ()

{

cout<<"\nThe simple Queue job are as follows....\n";

int temp;

for(temp=f;temp<=r;temp++)

{

cout<<"\t"<<simpq[temp];

}

}

//To delete job from the simple queue.

void spq::delprioQ()

{

if(isQempty())

cout<<"\nSorry Q is empty...\n";

else

{

djob=prioq[f];

f=f+1;

cout<<"\nDeleted job is: "<<djob;

}

}

//To show all the jobs from PrioQueue.

void spq::showprioQ()

{

cout<<"\nThe priority Queue job are as follows....\n";

int temp;

for(temp=f;temp<=r;temp++)

{

cout<<"\t"<<prioq[temp];

}

}

//To add the jobs as per the priority.

void spq::prioqadd()

{

int t=0;

cout<<"\nEnter the job: ";

cin>>job;

if(isQfull())

cout<<"\nSorry!! Priority Queue is full...\n";

else

{

if(f==-1)

{

f=r=0; //initially when q is empty insert first job.

prioq[r]=job;

}

else if(job<prioq[r]) //Check the priority(Ascending) of incoming job if it is high

{

t=r;

while(job<prioq[t])//do until priority is high

{

prioq[t+1]=prioq[t]; //then shift all the jobs towards right

t=t-1; //decrement index to check another job.

}

t=t+1; //increment index

prioq[t]=job; //store job at its appropriate location.

r=r+1; //increment rear index by one

}

else

{

r=r+1; // as per the priority store in Q.

prioq[r]=job;

}

}

}

int main()

{

spq s1,s2; //object creation. s1 for simple Q and s2 for priority Q

int ch;

do

{

cout<< "\n\t!!!Operating System Job Queue!!!" << endl; // prints the msg.

cout<<"\n1.SimpleQ Add\_Job\n2.SimpleQ Del\_Job\n3.Show SimpleQ\n4.PrioQ Add\_Job\n5.PrioQ Del\_Job\n6.Show PrioQ";

cout<<"\nEnter Your Choice:";

cin>>ch;

switch(ch)

{

case 1:s1.simpqadd();break;//calling adding element in simple Q without priority.

case 2:s1.delsimpleQ();break;

case 3:s1.showsimpleQ();break;

case 4:s2.prioqadd();break;//calling adding element in priority Q with priority.

case 5:s2.delprioQ();break;

case 6:s2.showprioQ();break;

}

}while(ch!=7);

return 0;

}