EXERCISE (1-3)

#include<bits/stdc++.h>

using namespace std;

#include<stdlib.h>

#include <conio.h>

struct Node

{

int data;

struct Node\*next;

} \*Front=NULL,\*Rare=NULL;

void enqueue(int x)

{

struct Node\*P;

P=(struct Node\*)malloc(sizeof(struct Node));

P->data=x;

P->next=NULL;

if(Front==NULL && Rare==NULL)

{

Front=Rare=P;

}

else

{

Rare->next=P;

Rare=P;

}

cout<<"\nInsertion is Success!!!\n"<<endl;

cout<<"\n\t\tEnter any key to go to menu"<<endl;

getch();

system("cls");

}

void dequeue()

{

struct Node\* temp;

if(Front==NULL){cout<<"The Queue is empty."<<endl;}

else

{

temp=Front;

Front=Front->next;

if(Front == NULL){Rare = NULL;}

free(temp);

}

cout<<"\nDelete is Success!!!\n"<<endl;

cout<<"\n\t\tEnter any key to go to menu"<<endl;

getch();

system("cls");

}

void PRINT()

{

struct Node \*temp = Front;

cout<<"The Queue is:"<<endl;

while(temp)

{

cout<<temp->data<<endl;

temp=temp->next;

}

cout<<"\n\t\tEnter any key to go to menu"<<endl;

getch();

system("cls");

}

int main()

{

int x,y;

while(1)

{

cout<<"\*\*\*\*\*\* Queue \*\*\*\*\*\*"<<endl;

cout<<"\n\_\_\_\_\_MENU\_\_\_\_\_ "<<endl;

cout<<"1. Create Queue\n2. Delete node from Queue\n3. Print Queue\n4. Exit"<<endl;

cout<<"Enter your choice: ";

cin>>x;

system("cls");

switch(x)

{

case 1:

cout<<"Enter the value to be insert: ";

cin>>y;

enqueue(y);

break;

case 2:

dequeue();

break;

case 3:

PRINT();

break;

case 4:

exit(0);

default:

cout<<"\nWrong selection!!! Please try again!!!"<<endl;

}

}

return 0;

}

EXERCISE (4)

#include<bits/stdc++.h>

using namespace std;

#include<stdlib.h>

#include <conio.h>

#define MAX 10

int Queue1[MAX],Queue2[MAX];

int Front=0, Rare=-1,Total=0;

int Full()

{

if(Total==MAX){return 1;}

return 0;

}

void enqueue(int Queue[],int x)

{

if(Full()){cout<<"Queue is full"<<endl;}

else

{

Rare=(Rare+1)%MAX;

Queue[Rare]=x;

Total++;

}

}

void PRINT(int Queue[],int n)

{

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

{

cout<<Queue[i]<<" ";

}

cout<<endl;

}

int main()

{

int n,value;

cout<<"Enter the number of total inputs: ";

cin>>n;

cout<<"Enter the values one by one:"<<endl;

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

{

cin>>value;

enqueue(Queue1,value);

}

cout<<"The 1st Queue is:"<<endl;

PRINT(Queue1,n);

Front=0, Rare=-1,Total=0;

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

{

enqueue(Queue2,Queue1[i]);

}

cout<<"The 2nd Queue is:"<<endl;

PRINT(Queue2,n);

return 0;

}

EXERCISE (5)

#include<bits/stdc++.h>

using namespace std;

#include<stdlib.h>

#define MAX 100

int Queue[MAX],Front=0, Rare=-1,Total=0;

int Full()

{

if(Total==MAX){return 1;}

return 0;

}

int Empty()

{

if(Total==0){return 1;}

return 0;

}

void enqueue(char val)

{

if(Full()){cout<<"Queue is full"<<endl;}

else

{

Rare=(Rare+1)%MAX;

Queue[Rare]=val;

Total++;

}

}

char dequeue()

{

char val;

if(Empty())cout<<"No input here"<<endl;

else

{

val = Queue[Front];

if(Front == Rare)

Front=Rare=-1;

else

Front++;

}

return val;

}

int main()

{

string line;

cout<<"Enter the string: ";

getline(cin,line);

for(int i=0;i<line.length();i++)

{

if(line[i]!= ' ')

{

enqueue(line[i]);

}

}

for(int i=Front;i<=Rare;i++)

{

line[i]=dequeue();

cout<<line[i];

}

cout<<endl;

return 0;

}

EXERCISE (6)

#include<bits/stdc++.h>

using namespace std;

#include<stdlib.h>

#define MAX 100

int Queue[MAX],Front=0, Rare=-1,Total=0;

struct Node

{

int data;

struct Node\*next;

}\*top=NULL;

void push(int value)

{

struct Node \*p;

p=(struct Node\*)malloc(sizeof(struct Node));

p->data = value;

if(top==NULL)

{p->next = NULL;}

else

{p->next = top;}

top=p;

}

void enqueue(int x)

{

Rare=(Rare+1);

Queue[Rare]=x;

Total++;

}

void pop()

{

if(top == NULL) cout<<"\nStack is Empty!!!"<<endl;

else

{

struct Node \*t = top;

enqueue(t->data);

top = t->next;

free(t);

}

}

void PRINT(int n)

{

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

{

cout<<Queue[i]<<" ";

}

cout<<endl;

}

int main()

{

int n,value;

cout<<"Enter the number of total inputs: ";

cin>>n;

cout<<"Enter the values in the stack one by one:"<<endl;

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

{

cin>>value;

push(value);

}

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

{

pop();

}

cout<<endl;

cout<<"The stored values in the Queue:"<<endl;

PRINT(n);

return 0;

}

EXERCISE (7)

#include<bits/stdc++.h>

using namespace std;

#include<stdlib.h>

#define MAX 100

int Queue[MAX],Front=0, Rare=-1,Total=0;

int Full()

{

if(Total==MAX) {return 1;}

return 0;

}

int Empty()

{

if(Total==0) {return 1;}

return 0;

}

void enqueue(char val)

{

if(Full())cout<<"Queue is full"<<endl;

else

{

Rare=(Rare+1)%MAX;

Queue[Rare]=val;

Total++;

}

}

void dequeue(int n)

{

int val;

if(Empty())cout<<"No input here"<<endl;

else

{

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

{

val = Queue[Front];

cout<<val<<" ";

Front++;

}

}

}

int main()

{

int n;

cout<<"Enter the number of total inputs: ";

cin>>n;

cout<<endl;

cout<<"Enter the values in the queue one by one:"<<endl;

int a[n],x=0;

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

{

cin>>a[i];

if(a[i]>=0)

{

enqueue(a[i]);

x++;

}

}

cout<<endl;

cout<<"After removing the values from the queue:"<<endl;

dequeue(x);

return 0;

}