// Program for implementation of stack using two queue

#include <stdio.h>

#include <stdlib.h>

#define size 3

int front1 = -1;

int rear1 = -1;

int front2 = -1;

int rear2 = -1;

int item;

int num = 0;

int cq1[size];

int cq2[size];

void main() {

int choice;

while (1) {

printf("\n Main Menu \n");

printf("1. Push\n 2. Pop\n 3. Display\n 4. Exit\n");

printf("Enter your choice: ");

scanf("%d", &choice);

switch (choice) {

case 1:

if (rear1 == size - 1) {

printf("Overflow - stack full\n");

}

else {

printf("Enter the item to push: ");

scanf("%d", &item);

push(item);

}

break;

case 2:

if (front1 == -1) {

printf("Underflow - stack empty\n");

} else {

pop();

}

break;

case 3:

display();

break;

case 4:

exit(0);

default:

printf("\nInvalid choice\n");

}

}

}

void push(int item) {

enq1(item);

num++;

}

int pop() {

int del;

for (int i = 0; i < num - 1; i++) {

int element1 = deq1();

enq2(element1);

}

del = deq1();

printf("\nThe deleted element is %d\n", del);

num--;

// Re-insert elements from cq2 to cq1

for (int i = 0; i < num; i++) {

int element2 = deq2();

enq1(element2);

}

return del; // Return the deleted element

}

int enq1(int element) {

if (front1 == -1)

front1 = 0;

rear1 = (rear1 + 1) % size;

cq1[rear1] = element;

return 1;

}

void enq2(int element) {

if (front2 == -1)

front2 = 0;

rear2 = (rear2 + 1) % size;

cq2[rear2] = element;

}

int deq1() {

item = cq1[front1];

if (front1 == rear1)

front1 = rear1 = -1;

else

front1 = (front1 + 1) % size;

return item;

}

int deq2() {

item = cq2[front2];

if (front2 == rear2)

front2 = rear2 = -1;

else

front2 = (front2 + 1) % size;

return item;

}

void display()

{

if (front1 == -1)

{

printf("Stack is empty\n");

}

else {

int i;

printf("Stack elements:\n");

for (i = rear1; i != front1; i = (i - 1+size) % size)

{

printf("%d\n", cq1[i]);

}

printf("%d\n", cq1[i]);

}

}

/\* Output:

Main Menu

1. Push

2. Pop

3. Display

4. Exit

Enter your choice: 1

Enter the item to push: 12

Main Menu

1. Push

2. Pop

3. Display

4. Exit

Enter your choice: 1

Enter the item to push: 32

Main Menu

1. Push

2. Pop

3. Display

4. Exit

Enter your choice: 1

Enter the item to push: 45

Main Menu

1. Push

2. Pop

3. Display

4. Exit

Enter your choice: 1

Enter the item to push: 63

Main Menu

1. Push

2. Pop

3. Display

4. Exit

Enter your choice: 1

Overflow - stack full

Main Menu

1. Push

2. Pop

3. Display

4. Exit

Enter your choice: 2

The deleted element is 63

Main Menu

1. Push

2. Pop

3. Display

4. Exit

Enter your choice: 3

Stack elements:

45

32

12

Main Menu

1. Push

2. Pop

3. Display

4. Exit

Enter your choice: 4 \*/