// C program to implement Queue using two stack.

#include<stdio.h>

#include<stdlib.h>

void push1(int);

void push2(int);

int pop1();

int pop2();

void enqueue();

void dequeue();

void display();

void create();

int stack1[150], stack2[150];

int top1 = -1, top2 = -1;

int count = 0;

int main()

{

int choice;

printf("\nQUEUE USING STACKS IMPLEMENTATION\n\n");

printf("\n1.ENQUEUE");

printf("\n2.DEQUEUE");

printf("\n3.DISPLAY");

printf("\n4.EXIT");

printf("\n");

create();

while (1)

{

printf("\nEnter your choice : ");

scanf("%d", &choice);

switch (choice)

{

case 1:

enqueue();

break;

case 2:

dequeue();

break;

case 3:

display();

break;

case 4:

exit(0);

default:

printf("\nInvalid Choice\n");

}}}

void create()

{

top1 = top2 = -1;

}

void push1(int element)

{

stack1[++top1] = element;

}

int pop1()

{

return(stack1[top1--]);

}

void push2(int element)

{

stack2[++top2] = element;

}

int pop2()

{

return(stack2[top2--]);

}

void enqueue()

{

int data, i;

printf("Enter the data : ");

scanf("%d", &data);

push1(data);

count++;

}

void dequeue()

{

int i;

for (i = 0;i <= count;i++)

{

push2(pop1());

pop2();

count--;

}

for (i = 0;i <= count;i++)

{

push1(pop2());

}

}

void display()

{

int i;

if(top1 == -1)

{

printf("\nEMPTY QUEUE\n");

}

else

{

printf("\nQUEUE ELEMENTS : ");

for (i = 0;i <= top1;i++)

{

printf(" %d ", stack1[i]);

}

printf("\n");

}

}

Output:

QUEUE USING STACK IMPLEMENTATION

1.ENQUEUE

2.DEQUEUE

3.DISPLAY

4.EXIT

Enter your choice : 1

Enter your data : 23

Enter your choice : 1

Enter your data : 26

Enter your choice : 1

Enter your data : 50

QUEUE ELEMENTS : 23 26 50

Enter your choice : 2

Enter your choice : 3

QUEUE ELEMENTS :23

Enter your choice : 4