DSA Lab 2

Selvakumar G (22MAI1004)

FIFO method using Queue with its operation in c.

#include <stdio.h>

#include <stdlib.h>

#define MAX 100

int queue[MAX];

int front = -1;

int rear = -1;

void enqueue(int);

int dequeue();

int isFull();

int isEmpty();

void display();

int main()

{

int choice, item;

while (1)

{

printf("1. Insert\n");

printf("2. Delete\n");

printf("3. Display\n");

printf("4. Quit\n");

printf("Enter your choice : ");

scanf("%d", &choice);

switch (choice)

{

case 1:

printf("Input the element for insertion in queue : ");

scanf("%d", &item);

enqueue(item);

break;

case 2:

item = dequeue();

printf("Deleted element is %d\n", item);

break;

case 3:

display();

break;

case 4:

exit(1);

default:

printf("Wrong choice\n");

}

}

return 0;

}

void enqueue(int item)

{

if (isFull())

{

printf("Queue Overflow \n");

return;

}

if (front == -1)

{

front = 0;

}

rear = rear + 1;

queue[rear] = item;

}

int dequeue()

{

int item;

if (isEmpty())

{

printf("Queue Underflow \n");

exit(1);

}

item = queue[front];

front = front + 1;

return item;

}

int isEmpty()

{

if (front == -1 || front > rear)

{

return 1;

}

else

{

return 0;

}

}

int isFull()

{

if (rear == MAX - 1)

{

return 1;

}

else

{

return 0;

}

}

void display()

{

int i;

if (isEmpty())

{

printf("Queue is empty \n");

return;

}

printf("Queue is : \n");

for (i = front; i <= rear; i++)

{

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

}

printf(" \n");

}

Output:

