Dewakar kotwal

Roll no:26

#include <stdio.h>

int Q[100], FRONT = -1, REAR = -1, i, n, x, choice;

void insert();

void delete ();

void display();

void main()

{

printf("\t WELCOME to implementation of QUEUE using array !! \n");

printf("Enter the size of Queue (Maximum size = 100): ");

scanf("%d", &n);

do

{

printf("\n Queue Operation available: \n");

printf("\t1.Insert \t2.Delete \t3.Display \t4.Exit \n");

printf("\n Enter your choice: ");

scanf("%d", &choice);

switch (choice)

{

case 1:

insert();

break;

case 2:

delete ();

break;

case 3:

display();

break;

case 4:

printf("Exit: Program Finished !! ");

break;

default:

printf("Please enter a valid choice 1, 2, 3, 4 \n");

break;

}

} while (choice != 4);

}

// Function to INSERT element

void insert()

{

if (REAR >= n - 1)

{

printf(" Queue Overflow ! \n");

}

else

{

printf(" Enter the element to insert: ");

scanf("%d", &x);

REAR++;

Q[REAR] = x;

if (FRONT == -1)

{

FRONT = 0;

}

}

}

// Function to DELETE element

void delete ()

{

if (FRONT == -1)

{

printf(" Queue Underflow ! \n");

}

else

{

printf(" The deleted element is: %d \n", Q[FRONT]);

if (FRONT == REAR)

FRONT = REAR = -1;

else

FRONT++;

}

}

// Functiom to DISPLAY Queue

void display()

{

if (REAR < 0)

{

printf(" Queue is empty ! \n");

}

else

{

printf(" The elements in the Queue are: \n");

for (i = FRONT; i < n; i++)

{

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

}

printf("\n");

}

}