Page No.:	
Date:	Aonny

Experiment no 5

> Aim - A double - ended queue (deque) is a linear list in which additions & deletions may be made at either end. Obtain a data representation mapping a deque into a onedimensional array. Write C++ program to simulate deque with functions to add f delete elements from either end of the deque.

Pre-requisite:

Knowledge of Queve

Types of Queve

Knowledge of double ended queue & different operations that can be performed on it.

Diective:
To simulate deque with function to add & delete elements from either end of the deque.

Input: Size of array relements in the queue:

Result of deque with functions to add & delete elements from either end of the deque.

Theory:

Double - Ended Queve : (- :) ! :

A double-ended queue is an abstract data type similar to an simpler simple queue, it allows you to insert & delete from both sides means items can be added or deleted from the front or reas end.

	La heriment n. 5	
	Algorithm for Insertion at rear end	
Kina mag	Sim of double could great (degred) is a linear list in which abliffice of the	6
000-	Step-1: [check for overflow] if (sear = = MAX) Point ("Queue is Over flow");	retu
121	Step 2 [Insert element] else	
4	rear = rear +1; supple set to made at mande states	
	9 [sea 8] = no;	
	[Set rear and Front pointer]	
	Leser reas and room pointed	
	of rear = 0 supple of supplement	
	rear; If front =0	
ed on it.	Foont; Stefn 3: retvonis to all some show all of subling	
	Implemenation of Insertion at reas end.	
	to simulate degre with function to all a delete demonts from either a	
	Void add_items_rear()	
	q and a second s	
	int nun;	
	point ("In Enter Items to insest: "); Scanf ("/d", & num"); If (sear == MAX))
	Outcome:	
the don	printF("\n Queue is OverFlow"); return;	
1	3	
	else	
	4	
	rear ++; q[rear] = num; 9/ (rear ==0) rear =1; 4/ (Front ==0) front =14	F,
Samo	A dulle rental angue is an abstract data true similar to an sincilar	
6	Agosithm for Ansertion at front end	
	detal from the beat or seed of	
S	teh 1: [Check for the front position] if (front <= 1) oint ("Cannot add item at front end"); return;	
P	point ("Connet add item at brent end"); return:	
84	ef 2: [Insert at Front] else	
	vont = loont-1; g[Front]= no; Atch-3: Return	
111	10100 - 41 001-0 1 1/1.1 1010-1 - 1/0 - 0 11/01 - 0 - 10/04/0	-

Page No.:

KOUV

Page No.:	
Date:	Aonny

Implementation of Inseration at front end	Amplementation of deletion
Void add_items_front()	void Idate item-Front ()
d	
int num;	int runs if (front == 0)
print F ("In Enter item to insert:"); &	canf ("/d, & num);
	pointf" In Queue is Under
if(Frant <=1)	octor:
9	5
Printf ("\n Cannot add item at front end");	do
return;	3
3	num = of [root];
else (root == troo) A : (mun " a) A !:	Lite (1) Albertini
a constitution of the same of	Testing to the version of
Front; q[Front] = num;	· Front =0; rear =0;
26	(0 = 0.000 to = 0.000 t
1	
4	9116
	jo
Algorithm for Deletion from front end	Front+;
	ş
Step-1 [Check for front pointer] if front =0	· ·
point ("Queue is Under flow");	\$
return;	
Step-2 [Perform deletion] else	Gerither for deletion from
1 1 2 1	Sept : Thek for the res
print (" Deleted element is", no); [Set front	
If front = rear front =0; rear =0;	said ("Canat delete ridue at
else front = front +1;	appea.
Step-3:	Stop of Legion deletion lake
Return	Constant of the same
	I drek for the front and
The property of the party of th	THE MOST SITE OF WARM I

		Date:	Konn
The real Persons in column 2 is not the owner, where the persons is not the persons in column 2 is not the person in column	Implementation of deletion from front end	Amplementation	
	void delete_item_Front()	void all i tem	
	{	70	
THE PERSON NAMED IN	int rum; if (foort == 0)	int nun;	
	Enter item to invest: "); Sant ("Id" Coun) :	printe ("In	
	print F(" \n Queue is Under flow \n");		
	odven;	if (Front <=1)	
	3	3	
	senset all item at front end");	printf ("In 1)	
The second lines in the second	q _ [r +7.	Tetuón;	
The second lives	num = 9 [Front];	1	
1	froint F ("In Deleted items is 1.d In", num); If (Front == 0	ear) all	
	Front = 0; rear = 0; mur = [tros	37. + .3	
	3 (Mary Class	Front ; 4 [F	
	else		
	e e e e e e e e e e e e e e e e e e e		
	Front++; has trail most mitable	Agraphical for	
	3	9	
	2 contract if front:0	St. 1. [Check	
	4 is Mades (Pini)	paint (" Queux	
	AO .0 1 10 0 1	returni	
	Algorithm for deletion from rear end		
	Step-1: [Check for the year pointer]	no a flourt	
	opint ("Cannot delete value at rear end");	print (" Velet	
	actives.		
-	Step 2: Éperform deletion Jobe	else front =	
	no = 9(8000);	Stop 3:	
I	[Check for the front and rear pointer of Front = rear front = 0; rear = 0; esse	11401570	
	front = 0; rear = 0; esse		

Page No.:

rear = rear -1; point (" Deleted element is", no); Step3 - Return Implementation of Deletion from year end Void delete_item_ rear () Point F ("In Cannot delete item at new end In"); return; num = q[rear]; 4(Front = = rear) Front =0; rear =0; 3 elso read --; point [" In Deleted items is 1.d \n", num):

Conclusion:
By this way, we can perform operations on double ended queue.