

Name Shehryar

Roll No Bsf2000745

Class BS IT Evening

Semester Fourth

Subject Data Structure and Algorithm

Course code COMP2113

Teacher Name Dr. Aftab Akram

**Activity 1:**

How to pass an array to function through pointer and pass by reference

**Through pointer**

#include<iostream>

using namespace std;

int show(int \*newArr,int n){

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

cout<<"You Enter the value at index "<<i<<": "<<\*(newArr+i)<<endl;

}

}

int main()

{

int n=0;

cout<<"Please Enter the size of array: ";

cin>>n;

int arr[n]={0};

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

cout<<"Please the value at index "<<i<<": ";

cin>>arr[i];

}

cout<<endl;

show(arr,n);

}

**Through Pass by reference:**

#include<iostream>

using namespace std;

int show(int ar[],int n){

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

cout<<"You Enter the value at index "<<i<<": "<<ar[i]<<endl;

}

}

int main()

{

int n=0;

cout<<"Please Enter the size of array: ";

cin>>n;

int arr[n]={0};

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

cout<<"Please the value at index "<<i<<": ";

cin>>arr[i];

}

cout<<endl;

show(arr,n);

}

**Output:**

Please Enter the size of array: 4

Please the value at index 0: 1

Please the value at index 1: 2

Please the value at index 2: 5

Please the value at index 3: 43

You Enter the value at index 0: 1

You Enter the value at index 1: 2

You Enter the value at index 2: 5

You Enter the value at index 3: 43

**Activity 2:**

From recursion find factorial of any number

#include<iostream>

using namespace std;

int factorial(int f){

if(f<=1){

return 1;

}else{

return f \* factorial(f-1);

}

}

int main(){

int n;

cout<<"please enter the number ";

cin>>n;

cout<<factorial(n);

return 0;

}

**Output:**

please enter the number 5

120

**Activity 3:**

Write a recursive function to find sum of first n even numbers

#include<iostream>

using namespace std;

int SumOfEven(int n){

if(n==1){

return 2;

}else{

return 2\*n +(SumOfEven(n-1));

}

}

int main(){

cout<<SumOfEven(6);

return 0;

}

**Output:**

42

**Activity 4:**

Put any random value any random index through array

#include <iostream>

using namespace std;

int arr[5]={0};

int randStore(){

int index;

string user;

cout << "Enter the index : ";

cin >> index;

//This condition check index are in bound or not

if(index<5)

{

cout << "\nAre you sure you enter the value index: "<<index<<". if (Y/N)";

cin>>user;

if(user=="y"){

true;

}else if(user=="n"){

cout<<"Again Enter index: ";

cin>>index;

}

}else{

cout<<"Please Enter the index in between 0 to 4"<<endl;

cout<<"Again Enter index: ";

cin>>index;

}

if(arr[index]!=0)

{

cout << "Value already exit at index "<<index <<". Try another index: ";

cin >> index;

}

cout << "Enter Value at "<<index<<": ";

cin >> arr[index];

}

void print(){

for(int i=0;i<5;i++){

cout<<"You Enter the value at index "<<i<<arr[i]<<endl;

}

}

int main()

{

int option,index=0,i,getter;

string user;

do{

cout<<endl<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<"\n\t\* \*";

cout<<"\n\t\* 1.value at random index \*";

cout<<"\n\t\* 2.value at each index from \*";

cout<<"\n\t\* user if user want to exit \*";

cout<<"\n\t\* from array press 0 \*";

cout<<"\n\t\* \*";

cout<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<"\n choose option: ";

cin>>option;

switch(option){

// case 1 for if user want to enter the value at specific index

case 1:

do{

cout<<endl<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<"\n\t\* \*";

cout<<"\n\t\* 1.Enter 1 if Enter value \*";

cout<<"\n\t\* 2.Enter 0 if Exit \*";

cout<<"\n\t\* \*";

cout<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<endl;

cout<<"Select: ";

cin >>getter;

switch(getter){

case 1:

randStore();

index++;

break;

case 0:

exit(0);

break;

default:

cout<<"\nPlease enter the valid index";

}

}while(index<5);

break;

// case 2, in this case user enter the value in each array index and if user want to exit

// press 0

case 2:

cout<<"\nPLease Enter the value in Array."<<endl;

cout<<"if user want to exit from array press 0.\n"<<endl;

for(i=0;i<10;i++){

cout<<"Please Enter the value at index "<<i<<": ";

cin>>arr[i];

// this condition for that if user want to exit from array by compare value if user

// press 0 this exit;

if(arr[i]==0){

cout<<"\nAre you sure you want to exit from array press if(y/n): ";

cin>>user;

if(user=="y"){

break;

}else if(user=="n"){

for(i;i<10;i++){

cout<<"Please Enter the value at index "<<i<<": ";

cin>>arr[i];

}

}

break;

}

}

cout<<endl;

for(int i=0;i<10;i++){

cout<<"You Enter at index"<<i<<": "<<arr[i]<<endl;

}

break;

default:

if(option!=2){

cout<<"\nPlease enter the valid number"<<endl;

}

}

}while(option<3);

print();

return 0;

}

**Output:**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.value at random index \*

\* 2.value at each index from \*

\* user if user want to exit \*

\* from array press 0 \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

choose option: 2

PLease Enter the value in Array.

if user want to exit from array press 0.

Please Enter the value at index 0: 6

Please Enter the value at index 1: 5

Please Enter the value at index 2: 4

Please Enter the value at index 3: 5

Please Enter the value at index 4: 25

Please Enter the value at index 5: 14

Please Enter the value at index 6: 25

Please Enter the value at index 7: 25

Please Enter the value at index 8: 45

Please Enter the value at index 9: 45

You Enter at index0: 6

You Enter at index1: 5

You Enter at index2: 4

You Enter at index3: 5

You Enter at index4: 25

You Enter at index5: 14

You Enter at index6: 25

You Enter at index7: 25

You Enter at index8: 45

You Enter at index9: 45

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.value at random index \*

\* 2.value at each index from \*

\* user if user want to exit \*

\* from array press 0 \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

choose option: 1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 1 if Enter value \*

\* 2.Enter 0 if Exit \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Select: 1

Enter the index : 55

Please Enter the index in between 0 to 4

Again Enter index: 68

Enter Value at 68: 25

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 1 if Enter value \*

\* 2.Enter 0 if Exit \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Select: 1

Enter the index : 54

Please Enter the index in between 0 to 4

Again Enter index: 25

Enter Value at 25: 5

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 1 if Enter value \*

\* 2.Enter 0 if Exit \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Select: 0

**Activity 5:**

How to use link List

#include<iostream>

using namespace std;

class linkedList{

public:

int data;

linkedList\* pointer;

};

void showLinkedList(linkedList \*ptr){

while(ptr!=NULL){

cout<<ptr->data<<endl;

ptr=ptr->pointer;

}

}

int main(){

linkedList \*firstHead = new linkedList();

linkedList \*secondHead = new linkedList();

firstHead->data=45;

firstHead->pointer=secondHead;

secondHead->data=78;

secondHead->pointer=NULL;

showLinkedList(firstHead);

return 0;

}

**Output:**

45

78

**Activity 6:**

Using Array and List Perform insert delete update and search operation in array

#include<iostream>

using namespace std;

int N,n;

bool isFull(){

if(N==n){

return true;

}else{

return false;

}

}

void insertAtStart(int ar[]){

if(isFull()){

cout<<"Memory Already full"<<endl;

}else{

int index=0,value;

cout<<"\nPlease Enter the value at "<<index<<": ";

cin>>value;

for(int i=N;i>index;i--){

ar[i]=ar[i-1];

}

ar[index]=value;

N++;

}

}

void insertAtBtw(int ar[]){

if(isFull()){

cout<<"Memory Already full"<<endl;

}else{

int index,value;

cout<<"\nPlease Enter the index: ";

cin>>index;

cout<<"\nPlease Enter the value: ";

cin>>value;

for(int i=N;i>index;i--){

ar[i]=ar[i-1];

}

ar[index]=value;

N++;

}

}

void insertAtEnd(int ar[]){

if(isFull()){

cout<<"Memory Already full"<<endl;

}else{

int value;

cout<<"\nPlease Enter the value: ";

cin>>value;

ar[N]=value;

N++;

}

}

void deleteAtStart(int ar[]){

int index=0;

for(int i=index;i<=N;i++){

ar[i]=ar[i+1];

}

ar[N]=0;

N--;

}

void deleteAtBtw(int ar[]){

int index;

cout<<"Please Enter the index where you delete the value: ";

cin>>index;

for(int i=index;i<=N;i++){

ar[i]=ar[i+1];

}

ar[N]=0;

N--;

}

void deleteAtEnd(int ar[]){

ar[N]=0;

N--;

}

void searchArray(int arr[]){

int value,counter=0,inc=0;

if(isFull()){

cout<<"memory is full";

}else{

cout<<"Please Enter the value: ";

cin>>value;

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

if(value==arr[i]){

counter=counter+1;

inc++;

}

}

if(inc==0){

cout<<"value not found";

}else{

cout<<"your Value "<<value<<" found: "<<counter<<" times."<<endl;

}

}

}

void updateArray(int arr[]){

int value,newValue,inc;

cout<<"Please Enter the value for which you update: ";

cin>>value;

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

if(value==arr[i]){

cout<<"Enter new Value: ";

cin>>newValue;

arr[i]=newValue;

inc++;

}

}

if(inc==0){

cout<<"value not found";

}else{

cout<<"update Array"<<endl;

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

cout<<arr[i]<<endl;

}

}

}

int main(){

N=0;

int n=20;;

if(n>=10){

int arr[n]={1,2,2,3,4,5,2,6,7,8,2,9,10};

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

if(arr[i]!=0){

N++;

continue;

}

break;

}

int get,i=0;

do{

cout<<endl<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<"\n\t\* \*";

cout<<"\n\t\* 1.Enter 0 if Exit \*";

cout<<"\n\t\* 2.Enter 1 if insert value \*";

cout<<"\n\t\* 3.Enter 2 if delete value \*";

cout<<"\n\t\* 4.Enter 3 if search value \*";

cout<<"\n\t\* 5.Enter 4 if update value \*";

cout<<"\n\t\* \*";

cout<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<endl;

cout<<"Please Select the option: ";

cin>>get;

switch(get){

case 0:

exit(0);

break;

case 1:

{

int i=0;

do{

cout<<endl<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<"\n\t\* \*";

cout<<"\n\t\* 1.Enter 0 if Exit \*";

cout<<"\n\t\* 2.Enter 1 if insert at First \*";

cout<<"\n\t\* 3.Enter 2 if insert at Between \*";

cout<<"\n\t\* 4.Enter 3 if insert at End \*";

cout<<"\n\t\* \*";

cout<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<endl;

int getter;

cout<<"Please select the value: ";

cin>>getter;

if(getter==0){

break;

}else if(getter==1){

cout<<"\tArray before inserting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

insertAtStart(arr);

cout<<"\n\tArray after inserting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

}else if(getter==2){

cout<<"\tArray before inserting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

insertAtBtw(arr);

cout<<"\n\tArray after inserting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

}else if(getter==3){

cout<<"\tArray before inserting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

insertAtEnd(arr);

cout<<"\n\tArray after inserting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

}else{

cout<<"Please Enter the valid Number"<<endl;

}

}while(i<2);

}

break;

case 2:

{

int i=0;

do{

cout<<endl<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<"\n\t\* \*";

cout<<"\n\t\* 1.Enter 0 if Exit \*";

cout<<"\n\t\* 2.Enter 1 if delete at First \*";

cout<<"\n\t\* 3.Enter 2 if delete at Between \*";

cout<<"\n\t\* 4.Enter 3 if delete at End \*";

cout<<"\n\t\* \*";

cout<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<endl;

int getter;

cout<<"Please select the value: ";

cin>>getter;

if(getter==0){

break;

}else if(getter==1){

cout<<"\tArray before deleting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

deleteAtStart(arr);

cout<<"\n\tArray after deleting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

}else if(getter==2){

cout<<"\tArray before deleting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

deleteAtBtw(arr);

cout<<"\n\tArray after deleting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

}else if(getter==3){

cout<<"\tArray before deleting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

deleteAtEnd(arr);

cout<<"\n\tArray after deleting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

}else{

cout<<"Please Enter the valid Number"<<endl;

}

}while(i<2);

}

break;

case 3:{

searchArray(arr);

}

break;

case 4:{

updateArray(arr);

}

break;

default:

cout<<"please Enter the valid number"<<endl;

}

}while(i<2);

}else{

cout<<"Please Enter the value greater than 10"<<endl;

}

return 0;

}

**Output:**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if insert value \*

\* 3.Enter 2 if delete value \*

\* 4.Enter 3 if search value \*

\* 5.Enter 4 if update value \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please Select the option: 1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if insert at First \*

\* 3.Enter 2 if insert at Between \*

\* 4.Enter 3 if insert at End \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the value: 2

Array before inserting

1,2,2,3,4,5,2,6,7,8,2,9,10,

Please Enter the index: 5

Please Enter the value: 56

Array after inserting

1,2,2,3,4,56,5,2,6,7,8,2,9,10

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if insert at First \*

\* 3.Enter 2 if insert at Between \*

\* 4.Enter 3 if insert at End \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the value: 0

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if insert value \*

\* 3.Enter 2 if delete value \*

\* 4.Enter 3 if search value \*

\* 5.Enter 4 if update value \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please Select the option: 2

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if delete at First \*

\* 3.Enter 2 if delete at Between \*

\* 4.Enter 3 if delete at End \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the value: 2

Array before deleting

1,2,2,3,4,56,5,2,6,7,8,2,9,10,

Please Enter the index where you delete the value: 6

Array after deleting

1,2,2,3,4,56,2,6,7,8,2,9,10,

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if delete at First \*

\* 3.Enter 2 if delete at Between \*

\* 4.Enter 3 if delete at End \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the value: 0

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if insert value \*

\* 3.Enter 2 if delete value \*

\* 4.Enter 3 if search value \*

\* 5.Enter 4 if update value \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please Select the option: 3

Please Enter the value: 5

value not found

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if insert value \*

\* 3.Enter 2 if delete value \*

\* 4.Enter 3 if search value \*

\* 5.Enter 4 if update value \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please Select the option: 4

Please Enter the value for which you update: 3

Enter new Value: 45

update Array

1

2

2

45

4

56

2

6

7

8

2

9

10

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if insert value \*

\* 3.Enter 2 if delete value \*

\* 4.Enter 3 if search value \*

\* 5.Enter 4 if update value \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please Select the option: 0

**Activity 7:**

Simple Queue operation mean Enqueue and dequeue

// simple Queue

#include<iostream>

using namespace std;

int n,N;

bool isFull(){

if(N==n){

return true;

}else{

return false;

}

}

bool isEmpty(){

if(N==0){

return true;

}else{

return false;

}

}

// Top at zero mean top fixed at zero index

void enQueue(int ar[]){

int val;

if(isFull()){

cout<<"Your queue is already full"<<endl;

}else{

cout<<"Please Enter the value";

cin>>val;

ar[N]=val;

N++;

}

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

cout<<ar[i]<<endl;

}

}

void deQueue(int ar[]){

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

ar[i]=ar[i+1];

}

ar[N-1]=0;

N--;

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

cout<<ar[i]<<endl;

}

}

// top at tail mean top fixed at last index;

void enQueueAtTailTop(int ar[]){

int val;

for(int i=N;i>0;i--){

ar[i]=ar[i-1];

}

cout<<"Please Enter the value";

cin>>val;

ar[0]=val;

N++;

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

cout<<ar[i]<<endl;

}

}

void deQueueAtTailTop(int ar[]){

ar[N-1]=0;

N--;

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

cout<<ar[i]<<endl;

}

}

int main(){

N=0,

n=20;

int arr[n]={1,2,3,4,5,6,78,2,8};

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

if(arr[i]!=0){

N++;

}

}

int i=0,getval;

do{

cout<<endl<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<"\n\t\* \*";

cout<<"\n\t\* 1.Enter 0 if Exit \*";

cout<<"\n\t\* 2.Enter 1 if perform top at head Queue \*";

cout<<"\n\t\* 3.Enter 2 if perform top at tail queue \*";

cout<<"\n\t\* \*";

cout<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<endl;

cout<<"Please select the option which your perform: ";

cin>>getval;

switch(getval){

case 0:

exit(0);

break;

case 1:

{

int i=0,get;

do{

cout<<endl<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<"\n\t\* \*";

cout<<"\n\t\* 1.Enter 0 if Exit \*";

cout<<"\n\t\* 2.Enter 1 if perform Enqueue operation at head \*";

cout<<"\n\t\* 3.Enter 2 if perform deQueue operation at head \*";

cout<<"\n\t\* \*";

cout<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<endl;

cout<<"Please select the option which your perform: ";

cin>>get;

if(get==0){

break;

}else if(get==1){

enQueue(arr);

}else if(get==2){

deQueue(arr);

}

}while(i<1);

break ;

}

case 2:

{

int i=0,get;

do{

cout<<endl<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<"\n\t\* \*";

cout<<"\n\t\* 1.Enter 0 if Exit \*";

cout<<"\n\t\* 2.Enter 1 if perform Enqueue operation at tail \*";

cout<<"\n\t\* 3.Enter 2 if perform deQueue operation at tail \*";

cout<<"\n\t\* \*";

cout<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<endl;

cout<<"Please select the option which your perform: ";

cin>>get;

if(get==0){

break;

}else if(get==1){

enQueueAtTailTop(arr);

}else if(get==2){

deQueueAtTailTop(arr);

}

}while(i<1);

}

}

}while(i<1);

return 0;

}

**Output:**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform top at head Queue \*

\* 3.Enter 2 if perform top at tail queue \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform Enqueue operation at head \*

\* 3.Enter 2 if perform deQueue operation at head \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 1

Please Enter the value36

1,2,3,4,5,6,78,2,8,36

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform Enqueue operation at head \*

\* 3.Enter 2 if perform deQueue operation at head \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 2

2,3,4,5,6,78,2,8,36

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform Enqueue operation at head \*

\* 3.Enter 2 if perform deQueue operation at head \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 1

Please Enter the value45

2,3,4,5,6,78,2,8,36,45

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform Enqueue operation at head \*

\* 3.Enter 2 if perform deQueue operation at head \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 0

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform top at head Queue \*

\* 3.Enter 2 if perform top at tail queue \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 0

**Activity 8:**

Circular Queue and Drift Queue

// array drifting

#include<iostream>

using namespace std;

int n=20;

int Front=-1,Back=-1;

bool isFull(){

if(Back==n-1){

return true;

}else{

return false;

}

}

bool isEmpty(){

if(Front==-1 && Back==-1){

return true;

}else{

return false;

}

}

void enQueue(int ar[]){

int val;

if(isFull()){

cout<<"Queue if full";

}else if(Front==-1 && Back==-1){

cout<<"Please Enter the value";

cin>>val;

Front=0,Back=0;

ar[Back]=val;

for(int i=Front;i<Back+1;i++){

cout<<"Your Elements in array "<<ar[i]<<endl;

}

}else{

cout<<"Please Enter the value";

cin>>val;

Back++;

ar[Back]=val;

for(int i=Front;i<Back+1;i++){

cout<<"Your Elements in array "<<ar[i]<<endl;

}

}

}

void deQueue(int ar[]){

if(isEmpty()){

cout<<"Your array is Empty";

}else if(Front == Back){

cout<<"DeQueElement is "<<ar[Front]<<endl;

Back=-1,Front=-1;

}else{

cout<<"DeQueElement is "<<ar[Front]<<endl;

Front++;

for(int i=Front;i<Back+1;i++){

cout<<"Your Elements in array "<<ar[i]<<endl;

}

}

}

int main(){

int arr[n]= {};

int i=0,getval;

do{

cout<<endl<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<"\n\t\* \*";

cout<<"\n\t\* 1.Enter 0 if Exit \*";

cout<<"\n\t\* 2.Enter 1 if perform top at head EnQueue \*";

cout<<"\n\t\* 3.Enter 2 if perform top at tail DEqueue \*";

cout<<"\n\t\* \*";

cout<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<endl;

cout<<"Please select the option which your perform: ";

cin>>getval;

switch(getval){

case 0:

exit(0);

break;

case 1:

enQueue(arr);

break ;

case 2:

deQueue(arr);

break;

default:

cout<<"please Enter the valid number"<<endl;

}

}while(i<1);

return 0;

}

**Output:**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform top at head EnQueue \*

\* 3.Enter 2 if perform top at tail DEqueue \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 1

Please Enter the value46

Your Elements in array 46

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform top at head EnQueue \*

\* 3.Enter 2 if perform top at tail DEqueue \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 1

Please Enter the value54

Your Elements in array 46

Your Elements in array 54

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform top at head EnQueue \*

\* 3.Enter 2 if perform top at tail DEqueue \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 1

Please Enter the value25

Your Elements in array 46

Your Elements in array 54

Your Elements in array 25

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform top at head EnQueue \*

\* 3.Enter 2 if perform top at tail DEqueue \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 2

DeQueElement is 46

Your Elements in array 54

Your Elements in array 25

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform top at head EnQueue \*

\* 3.Enter 2 if perform top at tail DEqueue \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 1

Please Enter the value45

Your Elements in array 54

Your Elements in array 25

Your Elements in array 45

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform top at head EnQueue \*

\* 3.Enter 2 if perform top at tail DEqueue \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 1

Please Enter the value98

Your Elements in array 54

Your Elements in array 25

Your Elements in array 45

Your Elements in array 98

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform top at head EnQueue \*

\* 3.Enter 2 if perform top at tail DEqueue \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 1

Please Enter the value35

Your Elements in array 54

Your Elements in array 25

Your Elements in array 45

Your Elements in array 98

Your Elements in array 35

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform top at head EnQueue \*

\* 3.Enter 2 if perform top at tail DEqueue \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 2

DeQueElement is 54

Your Elements in array 25

Your Elements in array 45

Your Elements in array 98

Your Elements in array 35

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform top at head EnQueue \*

\* 3.Enter 2 if perform top at tail DEqueue \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 0

**Activity 9:**

**Priority queue**

#include<iostream>

using namespace std;

int n=20,N;

bool isFull(){

if(N==n){

return true;

}else{

return false;

}

}

bool isEmpty(){

if(N==0){

return true;

}else{

return false;

}

}

void LowEnQueuePriority(int ar[]){

if(isFull()){

cout<<"array is full not performing enqueue operation"<<endl;

}else{

int val;

cout<<"please Enter the value for enQueue operation:";

cin>>val;

ar[N]=val;

N++;

}

}

void HighEnQueuePriority(int ar[]){

if(isFull()){

cout<<"array is full not performing enqueue operation"<<endl;

}else{

int val;

cout<<"please Enter the value for enQueue operation: ";

cin>>val;

for(int i=N;i>0;i--){

ar[i]=ar[i-1];

}

ar[0]=val;

N++;

}

}

void deQueue(int ar[]){

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

ar[i]=ar[i+1];

}

ar[N-1]=0;

N--;

}

int main(){

int arr[n] ={1,2,3,4,5,67,8,9};

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

if(arr[i]!=0){

N++;

}

}

int i=0,getVal;

do{

cout<<endl<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<"\n\t\* \*";

cout<<"\n\t\* 1.Enter 0 if Exit \*";

cout<<"\n\t\* 2.Enter 1 if perform top at head EnQueue \*";

cout<<"\n\t\* 3.Enter 2 if perform top at head DeQueue \*";

cout<<"\n\t\* \*";

cout<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<endl;

cout<<"Please select the option which your perform: ";

cin>>getVal;

switch(getVal){

case 0:

exit(0);

break;

case 1:

{

int i=0,getValue;

do{

cout<<endl<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<"\n\t\* \*";

cout<<"\n\t\* 1.Enter 0 if Exit \*";

cout<<"\n\t\* 2.Enter 1 if perform high priority enQueue \*";

cout<<"\n\t\* 3.Enter 2 if perform low priority enQueue \*";

cout<<"\n\t\* \*";

cout<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<endl;

cout<<"Please select the option which your perform: ";

cin>>getValue;

if(getValue==0){

break;

}else if(getValue==1){

cout<<"\tArray before inserting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

cout<<endl;

HighEnQueuePriority(arr);

cout<<"\n\tArray after inserting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

}else if(getValue==2){

cout<<"\tArray before inserting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

cout<<endl;

LowEnQueuePriority(arr);

cout<<"\n\tArray after inserting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

}else{

cout<<"Please Enter the valid number for EnQueue"<<endl;

}

}while(i<1);

}

break;

case 2:

cout<<"\tArray before deleting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

cout<<endl;

deQueue(arr);

cout<<"\n\tArray after deleting"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

break;

default:

cout<<"Please Enter the valid number";

break;

}

}while(i<1);

}

**Output:**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform top at head EnQueue \*

\* 3.Enter 2 if perform top at head DeQueue \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform high priority enQueue \*

\* 3.Enter 2 if perform low priority enQueue \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 1

Array before inserting

1,2,3,4,5,67,8,9,

please Enter the value for enQueue operation: 56

Array after inserting

56,1,2,3,4,5,67,8,9,

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform high priority enQueue \*

\* 3.Enter 2 if perform low priority enQueue \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 2

Array before inserting

56,1,2,3,4,5,67,8,9,

please Enter the value for enQueue operation:63

Array after inserting

56,1,2,3,4,5,67,8,9,63,

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform high priority enQueue \*

\* 3.Enter 2 if perform low priority enQueue \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 0

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform top at head EnQueue \*

\* 3.Enter 2 if perform top at head DeQueue \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 2

Array before deleting

56,1,2,3,4,5,67,8,9,63,

Array after deleting

1,2,3,4,5,67,8,9,63,

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if perform top at head EnQueue \*

\* 3.Enter 2 if perform top at head DeQueue \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please select the option which your perform: 0

**Activity 10:**

**Insertion Sort**

#include<iostream>

using namespace std;

void insertionSort(int ar[],int n){

cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Insertion Sort Start \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"before"<<endl;

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

cout<<ar[i]<<",";

}

for(int i=1;i<n;i++){

int j,temp=ar[i];

j=i-1;

while(j>=0&&ar[j]>temp){

ar[j+1]=ar[j];

j--;

}

ar[j+1]=temp;

}

cout<<"\nafter"<<endl;

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

cout<<ar[i]<<",";

}

cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Insertion Sort End \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

}

int main(){

int n=10;

int arr[n]={9,7,5,8,4,6,3,2,1,4};

insertionSort(arr,n);

return 0;

}

**Output:**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Insertion Sort Start \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

before

9,7,5,8,4,6,3,2,1,4,

after

1,2,3,4,4,5,6,7,8,9,

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Insertion Sort End \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Bubble Sort:**

#include<iostream>

using namespace std;

void bubbleSort(int ar[],int n){

cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Bubble Sort Start \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"before"<<endl;

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

cout<<ar[i]<<",";

}

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

int temp;

int flag=0;

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

if(ar[j]>ar[j+1]){

temp=ar[j];

ar[j]=ar[j+1];

ar[j+1]=temp;

flag=1;

}

}

if(flag==0){

break;

}

}

cout<<"\nAfter"<<endl;

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

cout<<ar[i]<<",";

}

cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Bubble Sort End \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

}

int main(){

int n=10;

int arr[n]={9,7,5,8,4,6,3,2,1,4};

bubbleSort(arr,n);

return 0;

}

**Output:**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Bubble Sort Start \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

before

9,7,5,8,4,6,3,2,1,4,

After

1,2,3,4,4,5,6,7,8,9,

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Bubble Sort End \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Selection Sort:**

#include<iostream>

using namespace std;

void selectionSort(int ar[],int n){

cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* selection Sort Start \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"before"<<endl;

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

cout<<ar[i]<<",";

}

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

int temp,minVal=i;

for(int j=i+1;j<n;j++){

if(ar[minVal]>ar[j]){

minVal=j;

}

}

temp=ar[i];

ar[i]=ar[minVal];

ar[minVal]=temp;

}

cout<<"\nAfter"<<endl;

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

cout<<ar[i]<<",";

}

cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* selection Sort End \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

}

int main(){

int n=10;

int arr[n]={9,7,5,8,4,6,3,2,1,4};

selectionSort(arr,n);

return 0;

}

**Ouput:**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* selection Sort Start \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

before

9,7,5,8,4,6,3,2,1,4,

After

1,2,3,4,4,5,6,7,8,9,

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* selection Sort End \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Activity No 11:**

**Stack Top at higher index and Top at lower index**

#include<iostream>

using namespace std;

int N,n;

bool isFull(){

if(N==n){

return true;

}else{

return false;

}

}

bool isEmpty(){

if(N==0){

return true;

}else{

return false;

}

}

void pushStackAtHigherIndex(int ar[]){

if(isFull()){

cout<<"Your Array is already Full you not enter further value"<<endl;

}else{

int val;

cout<<"Please push the value: ";

cin>>val;

ar[N]=val;

N++;

}

}

void popStackAtHigherIndex(int ar[]){

if(isEmpty()){

cout<<"You have no element in array for performing pop functionality"<<endl;

}else{

ar[N-1]=0;

N--;

}

}

void pushStackAtLowerIndex(int ar[]){

if(isFull()){

cout<<"Your Array is already Full you not enter further value"<<endl;

}else{

int val;

cout<<"Please Push the value: ";

cin>>val;

for(int i=N;i>0;i--){

ar[i]=ar[i-1];

}

ar[0]=val;

N++;

}

}

void popStackAtLowerIndex(int ar[]){

if(isEmpty()){

cout<<"You have no element in array for performing pop functionality"<<endl;

}else{

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

ar[i]=ar[i+1];

}

ar[N-1]=0;

N--;

}

}

int main(){

n=20;

int arr[n]={1,2,3,4,5,6,7,8,9,10};

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

if(arr[i]!=0){

N++;

}

}

int i=0,get;

do{

cout<<endl<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<"\n\t\* \*";

cout<<"\n\t\* 1.Enter 0 if Exit \*";

cout<<"\n\t\* 2.Enter 1 if you want to select atHigher Index \*";

cout<<"\n\t\* 3.Enter 2 if you want to select atLower Index \*";

cout<<"\n\t\* \*";

cout<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<endl;

cout<<"Please Select the option: ";

cin>>get;

switch(get){

case 0:

exit(0);

break;

case 1:

{

int i=0;

do{

cout<<endl<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Menu \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<"\n\t\* \*";

cout<<"\n\t\* 1.Enter 0 if Exit \*";

cout<<"\n\t\* 2.Enter 1 if you want to select push function AtHigher Index \*";

cout<<"\n\t\* 3.Enter 2 if you want to select pop function AtHigher Index \*";

cout<<"\n\t\* \*";

cout<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<endl;

int getVal;

cout<<"Please Select the option: ";

cin>>getVal;

if(getVal==0){

break;

}else if(getVal==1){

cout<<"\tArray before Push"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

cout<<endl;

pushStackAtHigherIndex(arr);

cout<<"\n\tArray after Push"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

cout<<endl;

}else if(getVal==2){

cout<<"\tArray before pop"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

cout<<endl;

popStackAtHigherIndex(arr);

cout<<"\n\tArray after pop"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

cout<<endl;

}else{

cout<<"Please Enter the valid Number"<<endl;

}

}while(i<1);

}

break;

case 2:

{

int i=0;

do{

cout<<endl<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Menu \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<"\n\t\* \*";

cout<<"\n\t\* 1.Enter 0 if Exit \*";

cout<<"\n\t\* 2.Enter 1 if you want to select push function AtLower Index \*";

cout<<"\n\t\* 3.Enter 2 if you want to select pop function AtLower Index \*";

cout<<"\n\t\* \*";

cout<<"\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout<<endl;

int getVal;

cout<<"Please Select the option: ";

cin>>getVal;

if(getVal==0){

break;

}else if(getVal==1){

cout<<"\tArray before Push"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

cout<<endl;

pushStackAtLowerIndex(arr);

cout<<"\n\tArray after Push"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

cout<<endl;

}else if(getVal==2){

cout<<"\tArray before pop"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

cout<<endl;

popStackAtLowerIndex(arr);

cout<<"\n\tArray after pop"<<endl;

cout<<"\t";

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

cout<<arr[i]<<",";

}

cout<<endl;

}else{

cout<<"Please Enter the valid Number"<<endl;

}

}while(i<1);

}

break;

default:

cout<<"please Enter the valid number"<<endl;

}

}while(i<0);

return 0;

}

**Output:**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Menu\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if you want to select atHigher Index \*

\* 3.Enter 2 if you want to select atLower Index \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please Select the option: 1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Menu \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if you want to select push function AtHigher Index \*

\* 3.Enter 2 if you want to select pop function AtHigher Index \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Please Select the option: 2

Array before pop

1,2,3,4,5,6,7,8,9,10,

Array after pop

1,2,3,4,5,6,7,8,9,

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Menu \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* 1.Enter 0 if Exit \*

\* 2.Enter 1 if you want to select push function AtHigher Index \*

\* 3.Enter 2 if you want to select pop function AtHigher Index \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*