





# VIRTUAL ANIMHOUSE

(DSA PROJECT 2022)

## Table of Content

- 1)TITLE OF THE PROJECT .
- 2)DECLARATION BY THE CANDIDATES.
- 3)PROBLEM STATEMENT AND OBJECTIVE.
- 4)APPROACH WITH WORKFLOW DIAGRAM
- 5)CODE SNIPPETS WITH MAJOR  
FUNCTIONALITIES.
- 6)OUTPUT SNAPSHOTS.
- 7)SOFTWARE REQUIREMENT.
- 8)CONCLUSION AND FUTURE WORK.
- 9)REFERENCES.


## DECLARATION BY THE CANDIDATE:-

1. ABHILASH TYAGI

**Enrollment number- 21104019**

## PROBLEM STATEMENT AND OBJECTIVE:

The Virtual Animhouse is a portal designed to enter the animal records in a secured virtual place where they are being taken care of by a master, where a master adds the animals , keeping record of their age and assigning them a IC number. This is designed using concepts of c++ stacks and the linked list . This system is based on the client's information idea. The customer can perform all errands here such as adding the animal data, retrieving their information once needed even in a sorted form and obviously , if



you changed your mind to add the animal in your virtual animhouse, you can remove that animal.

An animal house can be defined as an area in which the birds and animals are protected and are kept safely in their natural habitats , providing them a safe living and protecting them from the illegal activities like poaching and trafficking. So keeping in mind the importance of such initiatives , we came up with an idea of designing a virtual Animhouse, that is a portal to store the animal data in a proper manner so that other functionalities could be added in the future so that our initiative could reach its best potential. The Animal houses could be important for studies and research purposes so as to provide the animals with the best care. The main objective of the virtual animal house is to build an organic and optimal system of interaction for adding the animal data keeping record and retrieving them for further study and functionalities.

## **MAJOR FUNCTIONALITIES:**

## ADD THE ANIMAL DATA:

The client has to insert his/her choice first in order to proceed. There will be 7 choices which are Sort\_by IC Number Sort\_byName,Sort\_byage,Search, Display, removal and Exit.

```

=====
                                @@@@@@@@@@ HELLO MASTERR!!! @@@@@@@@@@
                                @@@@@@  WELCOMEE TO YOUR VIRTUAL ANIMAL SANCTURYY!!  @@@@@@@@@@

                                Menu:
                                1. Add your Animal here
                                2. Sort_by IC Number
                                3. Sort_by Name
                                4. Sort_byage
                                5. Search
                                6. Display
                                7. Exit
                                8. remove

Enter your choice> 1
Enter IC number of the animal: |
```

## SORTING BY IC NUMBER:

If The client will select choice 2, the software will ask for an animal IC number which will be allotted to one particular animal. The IC number will be unique for all animals of the sanctuary.

```

=====
                                HELLO MASTERR!!!
=====

        WELCOMEE TO YOUR VIRTUAL ANIMAL SANCTURY!!

Menu:
1. Add your Animal here
2. Sort_by IC Number
3. Sort_by Name
4. Sort_byage
5. Search
6. Display
7. Exit
8. remove

Enter your choice> 1
Enter IC number of the animal:234
Enter name of animal :
```

## SORT BY NAME:

Now , after giving the IC number the software will ask for an animal name .The client will allot a name to each animal of the sanctuary. The client can allot one same name to different animals.

```

=====
                                HELLO MASTERR!!!
=====

        WELCOMEE TO YOUR VIRTUAL ANIMAL SANCTURY!!

Menu:
1. Add your Animal here
2. Sort_by IC Number
3. Sort_by Name
4. Sort_byage
5. Search
6. Display
7. Exit
8. remove

Enter your choice> 1
Enter IC number of the animal:234
Enter name of animal :sheera
```

## SORT BY AGE :

The client can also put the age of the animal. The client can go to choice 4 and can input the age of the animals of sanctuary.

```
=====
                                @@@@@@@@@@ HELLO MASTERR!!! @@@@@@@@@@

                                @@@@@@  WELCOMEE TO YOUR VIRTUAL ANIMAL SANCTURY!! @@@@@@@@@@

                                Menu:
                                1. Add your Animal here
                                2. Sort_by IC Number
                                3. Sort_by Name
                                4. Sort_byage
                                5. Search
                                6. Display
                                7. Exit
                                8. remove

Enter your choice> 1
Enter IC number of the animal:234
Enter name of animal :sheera
Enter age of animal:23
```

## SEARCH:

This function allows the client to search the particular animal by its IC number . he /she can enter the IC number to retrieve the animal information.

```

          @@@@@@ WELCOMEE TO YOUR VIRTUAL ANIMAL SANCTURY!! @@@@@@@@@@

                                Menu:
          1. Add your Animal here
          2. Sort_by IC Number
          3. Sort_by Name
          4. Sort_byage
          5. Search
          6. Display
          7. Exit
          8. remove

Enter your choice> 1
Enter IC number of the animal:234
Enter name of animal :sheera
Enter age of animal:23

Enter your choice> 5
Enter name of the animal to be searched sheera

Animal found! Following is the information
IC Number: 234
Name: sheera
Age: 23

Enter your choice>

```

## REMOVE:

This function allows the client to delete any wrong entry or simply can remove the existing data.

```

          @@@@@@ WELCOMEE TO YOUR VIRTUAL ANIMAL SANCTURY!! @@@@@@@@@@

                                Menu:
          1. Add your Animal here
          2. Sort_by IC Number
          3. Sort_by Name
          4. Sort_byage
          5. Search
          6. Display
          7. Exit
          8. remove

Enter your choice> 1
Enter IC number of the animal:234
Enter name of animal :sheera
Enter age of animal:23

Enter your choice> 5
Enter name of the animal to be searched sheera

Animal found! Following is the information
IC Number: 234
Name: sheera
Age: 23

Enter your choice> 8
Removing your last animal u just added
Enter your choice>

```



## **TECHNOLOGIES USED:**

C++ is a general-purpose programming language created by Danish computer scientist Bjarne Stroustrup as an extension of the C programming language, or "C with Classes".

## **HARDWARE REQUIREMENTS:**

- Processor – 11th Gen Intel(R) Core <sup>™</sup> i5-1135G7 @2.40GHz 2.42 GHz • Installed RAM – 16GB
- System Type – 64 Bit Operating System, x64 based processor

## **SOURCE CODE:**

```
#include<iostream>
```

```
using namespace std;
```

```
//definition of structure of animal information
```



```
struct animal
```

```
{
```

```
int IC_Number;
```

```
string Name;
```

```
int age;
```

```
};
```

```
//definition of node of linked list
```

```
struct node
```

```
{
```

```
animal data;
```

```
struct node* next;
```

```
};
```

```
//definition of class stack
```

  
class stack

{

private:struct node\*top;//to store top position of stack//to swap the data of two structures of animal typevoid swap(animal &a, animal &b){int templ= a.IC\_Number;a.IC\_Number=b.IC\_Number;b.IC\_Number=templ;string tempn=a.Name;a.Name=b.Name;b.Name=tempn;



```
int tempa=a.age;
```

```
  a.age=b.age;
```

```
  b.age=tempa;
```

```
}
```

```
public :
```

```
//constructor to initialize top as NULL
```

```
stack()
```

```
{
```


```
top=NULL;
```

```
}
```

```
//function to prompt user for animal data and push it to stack
```

```
void push()
```

```
{
```



```
int num;
```

```
string name;
```

```
int age;
```

```
//prompt user to enter details
```

```
cout<<"Enter IC number of the animal:";
```

```
cin>>num;
```

```
cout<<"Enter name of animal :";
```

```
cin>>name;
```

```
cout<<"Enter age of animal:";
```

```
cin>>age;
```

```
cout<<endl;
```

```
//dynamically allocating a node and store address in pointer
```

```
node*t = new node;
```

  
//storing data in new nodet->data.IC\_Number=num;t->data.Name=name;t->data.age=age;t-> next = NULL;//if stack is emptyif(top==NULL){top=t;return;}//linking the new node to top element and changing top position//to new node



```
t->next= top;
```

```
top=t;
```

```
};
```

```
//to sort by IC number
```

```
void sort_byIC()
```

```
{
```

```
//if the stack is empty
```

```
if (top==NULL)
```


```
return ;
```

```
node*p = top;
```

```
node* last =NULL;
```

```
//to store whether elements are swapped or not
```

```
int swapped;
```



```
do

{

    swapped=0; // initially no elements are swapped

    p = top; // initializing p with top node

    //loop will stop when p reaches a node from where

    // all elements afterwards are sorted

    while( p->next!= last)

    {

        //swapping with next element if next element is

        // smaller than current element

        if(p->data.IC_Number> p->next->data.IC_Number)

        {

            swap(p->data,p->next->data);
```





```
_____ //as the element is swapped
```

```
_____ // flagged to 1(true)
```

```
_____ swapped =1;
```

```
_____ }
```

```
_____ //moving to next node
```

```
_____ p=p->next;
```

```
_____ }
```

```
_____ //assigning node to last which indicates that after last
```

```
_____ // all elements are sorted
```

```
_____ last=p;
```

```
_____ }
```

```
while(swapped!=0);//if all the elements are swapped then all elements
```

```
_____ // are sorted
```



```
}
```

```
//To sort by name
```

```
void sort_byname()
```

```
{
```

```
//If the stock is empty
```

```
if(top==NULL)
```

```
return;
```

```
node* p=top;
```


```
node* last= NULL;
```

```
int swapped;
```

```
do
```

```
{
```

```
swapped = 0;
```



```
p=top;
```

```
while (p->next!= last)
```

```
{
```

```
if (p->data.Name > p-> next->data.Name)
```

```
{
```

```
swap(p->data, p->next->data);
```

```
swapped=1;
```

```
}
```

```
p=p->next;
```

```
}
```

```
last=p;
```

```
}
```

```
while (swapped!=0);
```



```
}
```

```
//To sort by age
```

```
void sort_byage()
```

```
{
```

```
//If the stack is empty
```

```
if(top==NULL)
```

```
return;
```

```
node* p=top;
```


```
node* last=NULL;
```

```
int swapped;
```

```
do
```

```
{
```

```
swapped = 0;
```



```
p=top;
```

```
while (p->next != last)
```

```
{
```

```
if (p->data.age > p->next->data.age)
```

```
{
```

```
swap(p->data, p->next->data);
```

```
swapped = 1;
```

```
}
```

```
p = p->next;
```

```
}
```

```
last = p;
```

```
}
```

```
while (swapped!=0);
```



```
}
```

```
void u_changed_ur_mind()
```

```
{
```

```
string x="";
```

```
cout<<"Removing your last animal u just added";
```

```
cout<<endl;
```

```
node* temp=top;
```

```
if(top==NULL){
```

```
    cout<<"no animal added to remove";
```

```
}
```


```
else{
```

```
    x=top->data.Name;
```

```
    top=top->next;
```

```
}
```

```
}
```



```
void search()  
  
{  
    string name;  
  
    cout<<"Enter name of the animal to be searched ";  
  
    cin>>name;  
  
    cout<<endl;  
  
    node* p=top;  
  
//Traversing through stack  
  
while (p)  
  
{  
  
//Displaying information if name is found  
  
if (p->data.Name ==name)  
  
{
```

```
cout<<"Animal found! Following is the information\n";
```

```
cout<<"IC Number: "<< p->data.IC_Number << endl;
```

```
cout<<"Name: "<< p->data.Name << endl;
```

```
cout<<"Age: "<<p->data.age << endl << endl;
```

```
break;
```

```
}
```

```
p=p->next;
```

```
}
```

```
//p reached end means name not found
```

```
if(p==NULL)
```

```
{
```

```
    cout<<"Animal not found!\n";
```

```
}
```





```
}
```

```
void display()
```

```
{
```

```
//If the stock is empty
```

```
if (top== NULL)
```

```
{
```

```
cout<<"No animal is there\n";
```

```
return;
```

```
}
```

```
node* p=top;
```

### //Traversing through stack to display information

}

```
cout<<"Age: "<<p->data.age<<endl<<endl;
```

p=p->next;

}

}

 $\frac{1}{2}$ 

{

[illegible]

```
cout<<"  _^ ^ ^ ^ _  \_ _  _ _'  _^ ^ ^ ^ _"<<endl;
```

```
cout<<"          ),-----.(          ),-----.("<<endl;
```

```
cout<<"\t\t\t\t\t1. Add your Animal here\n\t\t\t\t\t2. Sort_by IC Number \n\t\t\t\t\t3. Sort_by Name\n";
```

```
cout<<"\t\t\t\t4. Sort byage\n\t\t\t\t5. Search\n\t\t\t\t6. Display\n\t\t\t\t7.  
Exit\n\t\t\t\t8. remove\n\n";
```

```
do
```

```
{
```

```
//Prompt user to enter choice
```

```
cout<<"Enter your choice> ";
```

```
cin>>ch;
```

```
switch(ch)
```

```
{
```

```
case 1: animal.push();
```

```
break;
```

```
case 2: animal.sort_byIC();
```

```
break;
```



case 3: animal.sort\_byname();

break;

case 4: animal.sort\_byage();

break;

case 5: animal.search();

break;

case 6: animal.display();

break;

case 7: cout << "exit"<<endl;

break;

case 8: animal.u\_changed\_ur\_mind();

break;

default: cout<<"Invalid choice\n";

}



```
}
```

```
while(ch!=7):
```

```
return 0;
```

```
};
```

**OUTPUT SNAPSHOT:**





```
                @@@@@@@@@@@@@ HELLO MASTERR!!! @@@@@@@@@@@@@
@          WELCOMEE TO YOUR VIRTUAL ANIMAL SANCTURYY!!          @

                                Menu:
1. Add your Animal here
2. Sort_by IC Number
3. Sort_by Name
4. Sort_byage
5. Search
6. Display
7. Exit
8. remove
```

## SOFTWARE REQUIREMENTS :

- Operating System – Windows 11.
- IDE.

## CONCLUSION AND FUTURE WORK:

We have accomplished the task of creating the logic for animal sanctuary and executing it successfully.

Over the board view , we can extend this project for a larger scale and bring it to a commercial purpose .

### **REFERENCE:**

- ❖ <https://www.techiedelight.com/stack-implementation-in-cpp/>
- ❖ <https://www.tutorialspoint.com/cplusplus-program-to-implement-singly-linked-list>