

DATA STRUCTURES & ALGORITHMS

Muhammad Anas

Sap Id: 55412

Lab 02

1. Write down a program in C++ that take an age of 10 students as an input from user and display the largest age of the student from an array.

```
#include <iostream>
using namespace std;
int main()
{
    int a[10]{};
    int maxage = 0, i, j;
    for (i = 0; i < 10; i++)
    {
        cout << "\n Enter the age of the student at index " << i << "
= ";
        cin >> a[i];
    }
    for (j = 0; j < 10; j++)
    {
        if (a[j] > maxage)
        {
            maxage = a[j];
        }
    }
}
```

```
cout << "\n" << maxage << " is the MAXIMUM AGE of the  
student" << endl;  
}
```

Output

```
Enter the age of the student at index 0 = 19  
Enter the age of the student at index 1 = 19  
Enter the age of the student at index 2 = 18  
Enter the age of the student at index 3 = 18  
Enter the age of the student at index 4 = 20  
Enter the age of the student at index 5 = 18  
Enter the age of the student at index 6 = 19  
Enter the age of the student at index 7 = 21  
Enter the age of the student at index 8 = 19  
Enter the age of the student at index 9 = 20  
21 is the MAXIMUM AGE of the student
```

-
2. Write down a program in C++ that take an input from user in three different arrays and then add the arrays and store them in another array. (Through Dynamic Array Concept).

```
#include <iostream>  
using namespace std;
```

```

int main() {
    int size;
    do {
        cout << "\n Enter the size of the arrays = ";
        cin >> size;
        if (size <= 0)
        {
            cout << "\n Error: Array size must be positive. Please try
again.\n";
        }
    } while (size <= 0);
    int* arr1 = new int[size];
    int* arr2 = new int[size];
    int* arr3 = new int[size];
    int* result = new int[size];
    cout << "\n Enter " << size << " elements for the first array =\n";
    for (int i = 0; i < size; i++)
    {
        cout << " ";
        cin >> arr1[i];
    }
    cout << "\n Enter " << size << " elements for the second array
=\n";
    for (int i = 0; i < size; i++)
    {
        cout << " ";
        cin >> arr2[i];
    }
    cout << "\n Enter " << size << " elements for the third array =\n";

```

```

for (int i = 0; i < size; i++)
{
    cout << " ";
    cin >> arr3[i];
}
for (int i = 0; i < size; i++)
{
    cout << " ";
    result[i] = arr1[i] + arr2[i] + arr3[i];
}
cout << "\n The sum of corresponding elements is =\n";
for (int i = 0; i < size; i++) {
    cout << " " << result[i] << " ";
}
cout << endl;
delete[] arr1;
delete[] arr2;
delete[] arr3;
delete[] result;
return 0;
}

```

Output

```
Enter the size of the arrays = 4
Enter 4 elements for the first array =
1
2
3
4
Enter 4 elements for the second array =
5
6
7
8
Enter 4 elements for the third array =
9
10
11
12
The sum of corresponding elements is =
15 18 21 24
```

3. Write a program for linear search using the concept of dynamic array (Note: Program should handle the situation if item is not in the list).

```
#include <iostream>
using namespace std;
int main()
{

    int* dArray;
    int size, target;
```

```

int result = -1;
cout << "\n Enter the size of the array = ";
cin >> size;
dArray = new int[size];
cout << "\n Enter " << size << " elements =\n";
for (int i = 0; i < size; i++)
{
    cout << " ";
    cin >> dArray[i];
}
cout << "\n Enter the element to search = ";
cin >> target;
for (int i = 0; i < size; i++)
{
    if (dArray[i] == target)
    {
        result = i;
        break;
    }
}
if (result != -1)
{
    cout << "\n Element " << target << " found at index " << result
<< endl;
}
else
{
    cout << "\n Element " << target << " not found in the array." <<
endl;
}

```

```
}  
delete[] dArray;  
return 0;  
}
```

Output

```
Enter the size of the array = 4  
  
Enter 4 elements =  
12  
20  
30  
40  
  
Enter the element to search = 30  
  
Element 30 found at index 2
```