

LAB - 2

Q1. Wap to create an Dynamic Array and display elements.

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int *ptr = NULL;
    int i, n = 0;
    printf("ENTER THE SIZE OF ARRAY: ");
    scanf("%d", &n);
    ptr = (int *)malloc(n * sizeof(int));
    printf("ENTER THE ELEMENTS: ");
    for (i = 0; i < n; ++i)
    {
        scanf("%d", &ptr[i]);
    }
    printf("~~~~~ Array elements are ~~~~\n");
    for (i = 0; i < n; ++i)
    {
        printf("%d\t", ptr[i]);
    }
    return 0;
}
```



```
ENTER THE SIZE OF ARRAY: 5
ENTER THE ELEMENTS: 4
3
5
2
1
~~~~~ Array elements are ~~~~
4      3      5      2      1
```

Q2. Wap to allocate memory for 5 no and display it by using malloc.

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int *ptr = NULL;
    int i, len = 5;
    ptr = (int *)malloc(len * sizeof(int));
    printf("ENTER THE ELEMENTS: ");
    for (i = 0; i < len; ++i)
    {
        scanf("%d", &ptr[i]);
    }
    printf("Elements stored using malloc:\n");
    for (i = 0; i < len; ++i)
    {
        printf("%d\t", ptr[i]);
    }
}
```

```

}
return 0;
}

```

```

ENTER THE ELEMENTS: 5
2
3
1
4
Elements stored using malloc:
5      2      3      1      4

```

Q3. Wap to allocate 5 no and find greatest no by using calloc.

```

#include <stdio.h>
#include <stdlib.h>
int main()
{
    int n;
    double *data;
    printf("Enter the total number of elements: ");
    scanf("%d", &n);
    data = (double *)calloc(n, sizeof(double));
    if (data == NULL)
    {
        printf("Error!!! memory not allocated.");
    }
    for (int i = 0; i < n; ++i)
    {
        printf("Enter number %d: ", i + 1);
        scanf("%lf", data + i);
    }
    for (int i = 1; i < n; ++i)
    {
        if (*data < *(data + i))
        {
            *data = *(data + i);
        }
    }
    printf("Largest number = %.2lf", *data);
    return 0;
}

```

```

Enter the total number of elements: 5
Enter number1: 1
Enter number2: 3
Enter number3: 5
Enter number4: 2
Enter number5: 4
Largest number = 5.00

```

Q4. Wap to add two no in c++.

```

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

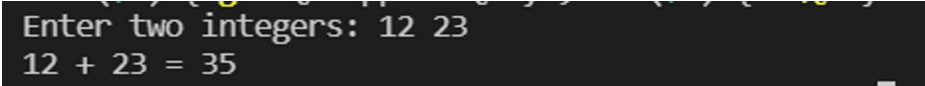
```

```

int firstNumber, secondNumber, sumOfTwoNumbers;

cout << "Enter two integers: ";
cin >> firstNumber >> secondNumber;
// sum of two numbers is stored in variable sumOfTwoNumbers
sumOfTwoNumbers = firstNumber + secondNumber;
// Print sum
cout << firstNumber << " + " << secondNumber << " = " << sumOfTwoNumbers;
return 0;
}

```

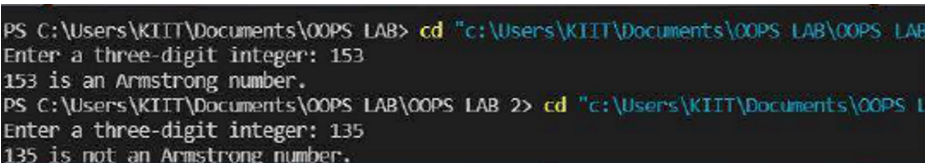


Q5. Wap to check no is armstrong or not of 3 digits no in C++.

```

#include <iostream>
using namespace std;
int main()
{
    int num, oNum, rmdr, res = 0;
    cout << "Enter a three-digit integer: ";
    cin >> num;
    oNum = num;
    while (oNum != 0)
    {
        rmdr = oNum % 10;
        res += rmdr * rmdr * rmdr;
        oNum /= 10;
    }
    if (res == num)
        cout << num << " is an Armstrong number.";
    else
        cout << num << " is not an Armstrong number.";
    return 0;
}

```



Q6. Wap to create student structure having roll, name, mark1, mark2, mark3 and display sum(), avg() using pointer to structure in C.

```

#include <stdio.h>
void main()
{
    struct student
    {
        int rollno;
        char name[20];
        int m1, m2, m3, total;
        float average;
    }
}

```

```

}
struct student s[20], t;
int i, j, n;
printf("\n Enter the number of students: ");
scanf("%d", &n);
for (i = 0; i < n; i++)
{
    printf("Enter the roll no: ");
    scanf("%d", &s[i].rollno);
    printf("\nEnter the name: ");
    scanf("%s", s[i].name);
    printf("\nEnter the mark 1: ");
    scanf("%d", &s[i].m1);
    printf("\nEnter the mark 2: ");
    scanf("%d", &s[i].m2);
    printf("\nEnter the mark 3: \n");
    scanf("%d", &s[i].m3);
    s[i].average = (s[i].m1 + s[i].m2 + s[i].m3) / 3;
}
for (i = 0; i < n - 1; i++)
{
    for (j = i + 1; j < n; j++)
    {
        if (s[i].average < s[j].average)
        {
            t = s[i];
            s[i] = s[j];
            s[j] = t;
        }
    }
}
printf("\n-----\n");
for (i = 0; i < n; i++)
{
    printf("\n-----\n");
    printf("\n rollno = %d", s[i].rollno);
    printf("\n name = %s", s[i].name);
    printf("\n mark1 = %d", s[i].m1);
    printf("\n mark2 = %d", s[i].m2);
    printf("\n mark3 = %d", s[i].m3);
    printf("\n average = %.2lf", s[i].average);
    printf("\n Total = %d", s[i].m1 + s[i].m2 + s[i].m3);
}
}

```

```

Enter the number of students: 2
Enter the roll no: 1

Enter the name: Anupam

Enter the marks 1:50
Enter the marks 2: 49
Enter the marks 3:
48
Enter the roll no: 2

Enter the name: 2

Enter the marks 1:45
Enter the marks 2: 46
Enter the marks 3:
47

```

```

-----
rollno = 1
name = Anupam
mark1 = 50
mark2 = 49
mark3 = 48
average = 49.00
Total = 147
-----

```

```

rollno = 2
name = 2
mark1 = 45
mark2 = 46
mark3 = 47
average = 46.00
Total = 138

```

Q7. Wap to create Emp structure having empno, ename, esal. Use input() to take input and display function for display it.

```

#include <stdio.h>
typedef struct
{
    char ename[30];
    int empno;
    int esal;
} Employee;
int main()
{
    int i, n = 2;
    Employee emp[n];
    printf("Enter %d Employee Details \n \n", n);
    for (i = 0; i < n; i++)
    {
        printf("Employee %d:- \n", i + 1);
        printf("Name:");
        gets(emp[i].ename);
        printf("Id:");
        scanf("%d", &emp[i].empno);
        printf("Salary:");
        scanf("%d", &emp[i].esal);
        printf("\n");
    }
}

```

```
    fflush(stdin);  
}  
printf("----- All Employees Details ----- \n");  
for (i = 0; i < n; i++)  
{  
    printf("Name \t:");  
    printf("%s \n", emp[i].ename);  
    printf("Id \t:");  
    printf("%d \n", emp[i].empno);  
    printf("Salary \t:");  
    printf("%d \n", emp[i].sal);  
    printf("\n");  
}  
return 0;  
}
```

```
Enter 2 Employee Details  
  
Employee 1:-  
Name: Anupam  
Id: 1  
Salary: 10000  
  
Employee 2:-  
Name: Moharana  
Id: 2  
Salary: 15000  
  
----- All Employees Details -----  
Name      : Anupam  
Id         : 1  
Salary     : 10000  
  
Name      : Moharana  
Id         : 2  
Salary     : 15000
```