

NAME: M.SRIJA

ROLL NO : CH.SC.U4CSE24126

DAA WEEK-1

1. write program to find sum on n natural numbers using user define function

```
//CH.SC.U4CSE24126
#include <stdio.h>
void sumofn(int n){
    int sum=0;
    for(int i=1;i<n+1;i++){
        sum+=i;
    }
    printf("%d",sum);
}
int main(){
    int n;
    printf("Enter the value for n \n");
    scanf("%d",&n);
    sumofn(n);
}
```

```
amma@amma23:~/DAA$ gcc -o 1 1.c
amma@amma23:~/DAA$ ./1
Enter the value for n
5
15amma@amma23:~/DAA$ gcc -o 2 2.c
```

#### JUSTIFICATION:

O(1) because the program uses only a fixed number of integer variables (sum, i, n).

2. write a program to find sum of squares of first n natural numbers

```
//CH.SC.U4CSE24126
#include <stdio.h>
int main(){
    int n,i,sum=0;
    printf("enter a number n:\n");
    scanf("%d",&n);
    for(i=1;i<=n;i++){
        sum+=i*i;
    }
    printf("the sum of %d natural numbers is %d",n,sum);
    return 0;
}
```

```
10amma@amma23:~/DAA$ gcc -o 2 2.c
amma@amma23:~/DAA$ ./2
enter a number n:
4
the sum of 4 natural numbers is 30
```

### JUSTIFICATION:

O(1) because it Uses only a fixed number of integer variables.

3. write a program sum of cubes of first n natural numbers

```
//CH.SC.U4CSE24126
#include <stdio.h>
int main(){
    int n,i,sum=0;
    printf("enter a number n:\n");
    scanf("%d",&n);
    for(i=1;i<=n;i++){
        sum+=i*i*i;
    }
    printf("the sum of %d natural numbers is %d",n,sum);
    return 0;
}
```

```
amma@amma23:~/DAA$ ./3
enter a number n:
3
the sum of 3 natural numbers is 36
```

### JUSTIFICATION:

O(1)

It will not make new arrays, only sum loop  
updates single variable

4. write a program to find factorial of given integer using recursion

```
//CH.SC.U4CSE24|126
#include <stdio.h>
int fact(int n){
if(n==0 || n==1){
return 1;
}
else{
return n*fact(n-1);
}
}

int main(){
int n;
printf("Enter the number:\n");
scanf("%d",&n);
int result=fact(n);
printf("%d",result);
}
```

```
amma@amma23:~/DAA$ gcc -o 4 4.c
amma@amma23:~/DAA$ ./4
Enter the number:
5
120amma@amma23:~/DAA$
```

JUSTIFICATION:

$O(n)$

Function calls itself until it completes the calculation

5. write a program for transposing a 3\*3 matrices

```
//CH.SC.U4CSE24126
#include <stdio.h>
int main() {
    int a[3][3], t[3][3];
    int i, j;
    printf("Enter the elements of the 3x3 matrix:\n");
    for(i = 0; i < 3; i++) {
        for(j = 0; j < 3; j++) {
            scanf("%d", &a[i][j]);
        }
    }
    for(i = 0; i < 3; i++) {
        for(j = 0; j < 3; j++) {
            t[j][i] = a[i][j];
        }
    }
    printf("The transpose of the matrix is:\n");
    for(i = 0; i < 3; i++) {
        for(j = 0; j < 3; j++) {
            printf("%d ", t[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

```
PS C:\Users\mrao0\OneDrive\Documents\C POINTERS\TREES> gcc 55.c -o tree.exe
PS C:\Users\mrao0\OneDrive\Documents\C POINTERS\TREES> ./tree.exe
Enter the elements of the 3x3 matrix:
1
2
3
4
5
6
7
8
9
The transpose of the matrix is:
1 4 7
2 5 8
3 6 9
```

O(1) because it Uses fixed-size 3×3 arrays and a few loop variables.

6. write a program to find fibonacci series.

```
//CH.SC.U4CSE24126
#include <stdio.h>

int main() {
    int n, a = 0, b = 1, c;

    printf("Enter how many terms you want:\n");
    scanf("%d", &n);

    printf("Fibonacci Series:\n");

    for(int i = 1; i <= n; i++) {
        printf("%d ", a);
        c = a + b;
        a = b;
        b = c;
    }

    return 0;
}
```

```
PS C:\Users\mrao0\OneDrive\Documents\C POINTERS\TREES> gcc 56.c -o tree.exe
PS C:\Users\mrao0\OneDrive\Documents\C POINTERS\TREES> ./tree.exe
Enter how many terms you want:
7
Fibonacci Series:
0 1 1 2 3 5 8
```

## JUSTIFICATION:

O(1) This code uses fixed no.of variables to store integers.

We are not storing whole fibonacci numbers in array