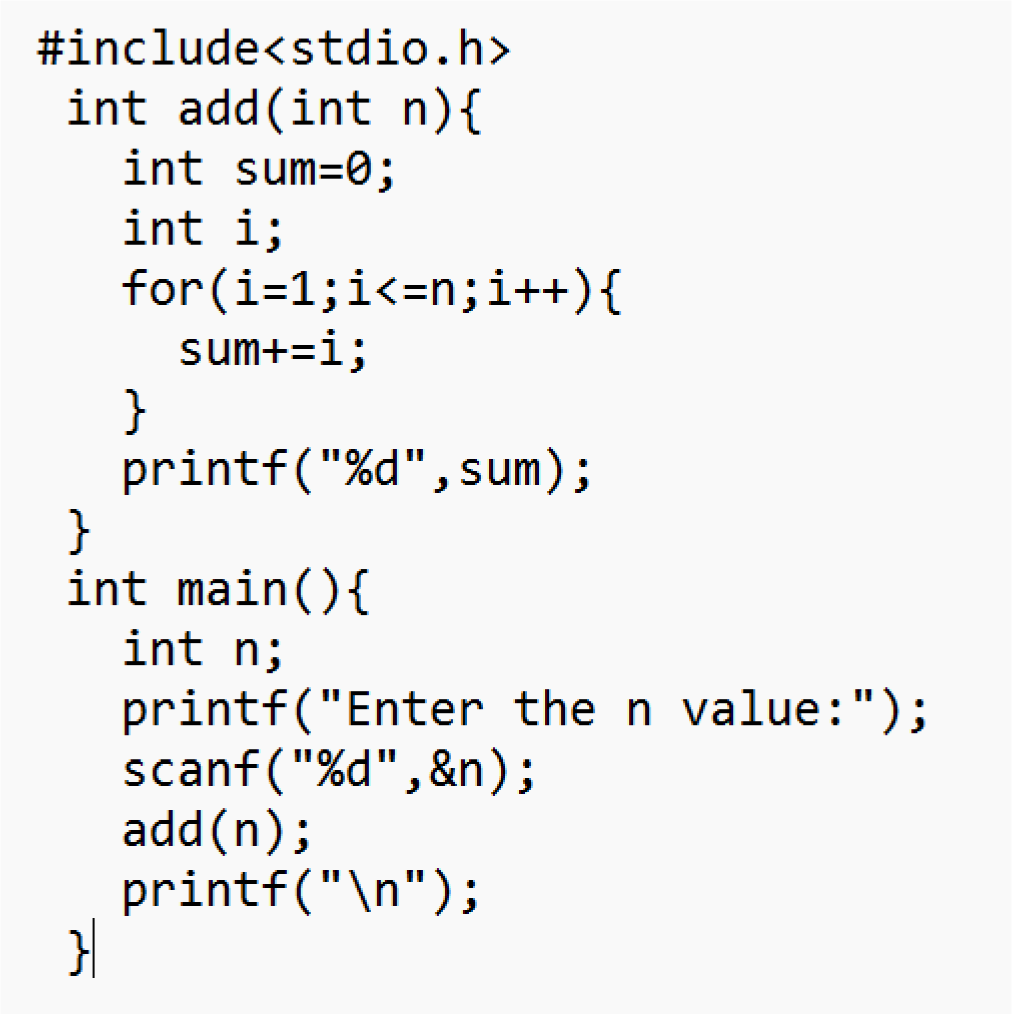
WEEK-1

Name: T.vasantha

Roll no: CH.SC.U4CSE24147

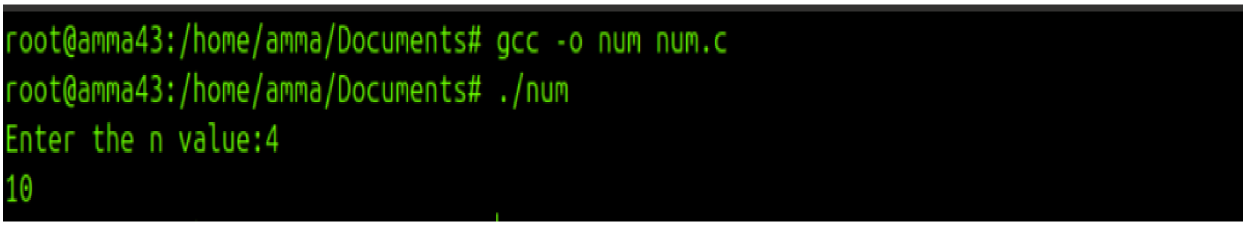
1)Problem statement: Write a program to write sum of first n natural numbers using user defined function.

Code:



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

Output:

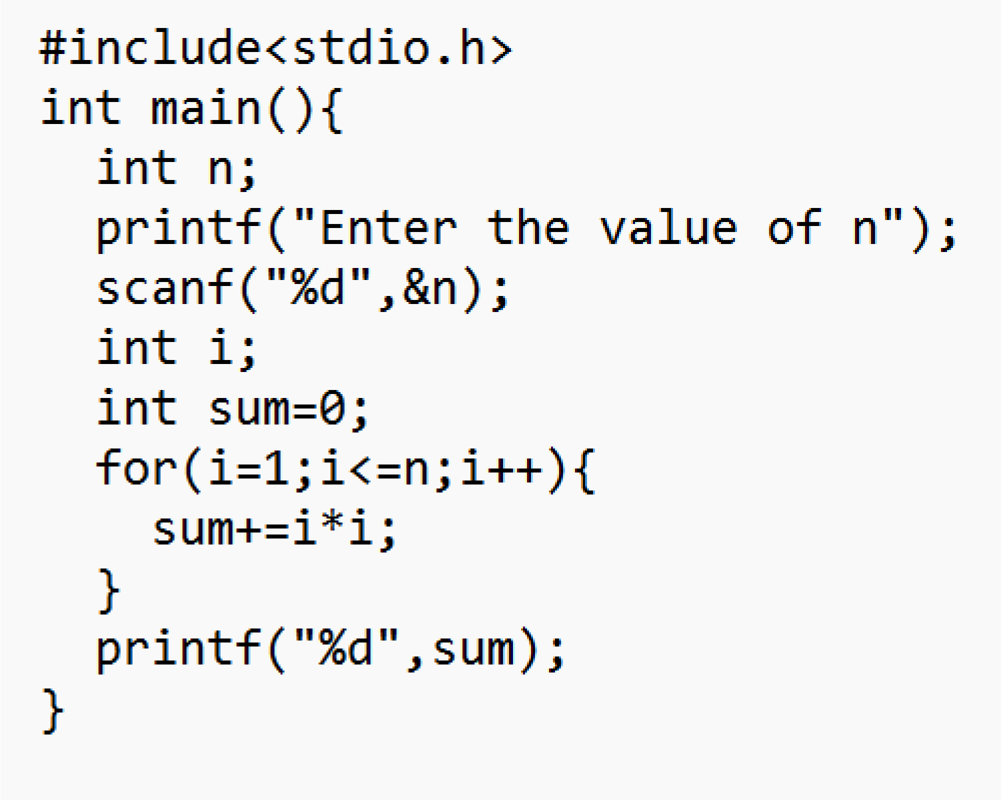


```
root@amma43:/home/amma/Documents# gcc -o num num.c
root@amma43:/home/amma/Documents# ./num
Enter the n value:4
10
```

Space complexity: It uses only fixed size variable and it uses only one and it keeps changing for every iteration and no new variable is used so the space complexity for this code is **0(1)**

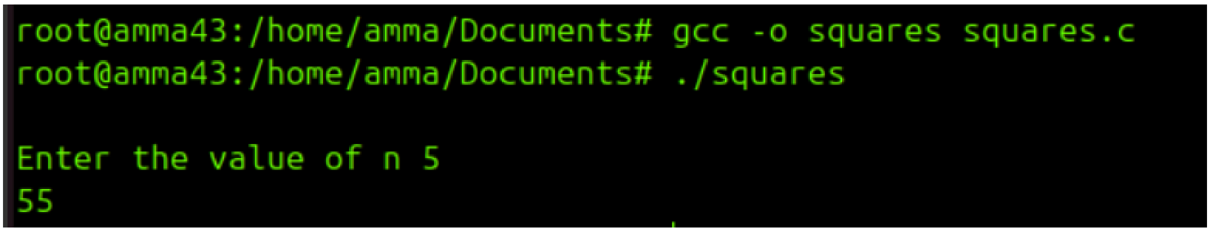
2)Problem statement: Write a program to find sum of squares of the first natural numbers.

Code:



```
#include<stdio.h>
int main(){
   int n;
   printf("Enter the value of n");
   scanf("%d",&n);
   int i;
   int sum=0;
   for(i=1;i<=n;i++){
      sum+=i*i;
   }
   printf("%d",sum);
}
```

Output:



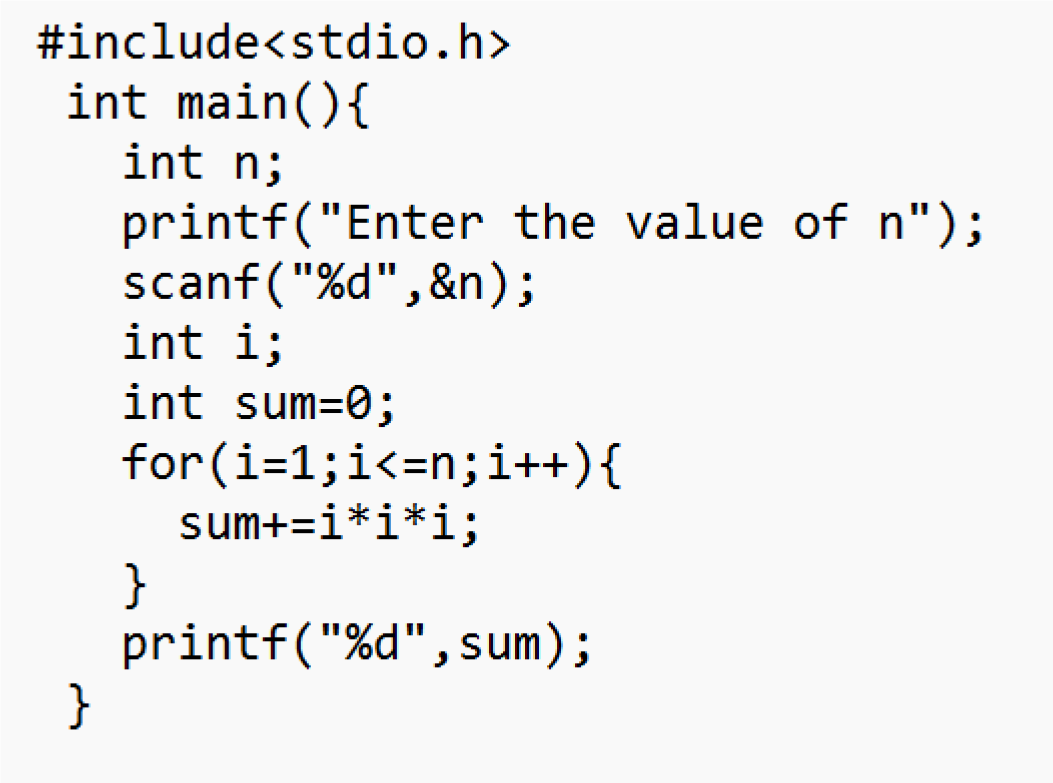
```
root@amma43:/home/amma/Documents# gcc -o squares squares.c
root@amma43:/home/amma/Documents# ./squares

Enter the value of n 5
55
```

Space complexity: Same as above problem ,It uses only fixed size variable and it uses only one and it keeps changing for every iteration and no new variable is used so the space complexity for this code is **0(1)**

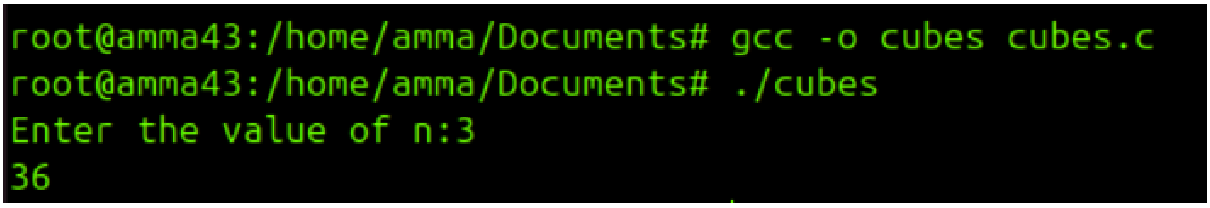
3)Problem statement: Write a program to find sum of cubes of the first natural numbers.

Code:



```
#include<stdio.h>
 int main(){
   int n;
   printf("Enter the value of n");
   scanf("%d",&n);
   int i;
   int sum=0;
   for(i=1;i<=n;i++){
     sum+=i*i*i;
   }
   printf("%d",sum);
 }
```

<u>Output:</u>

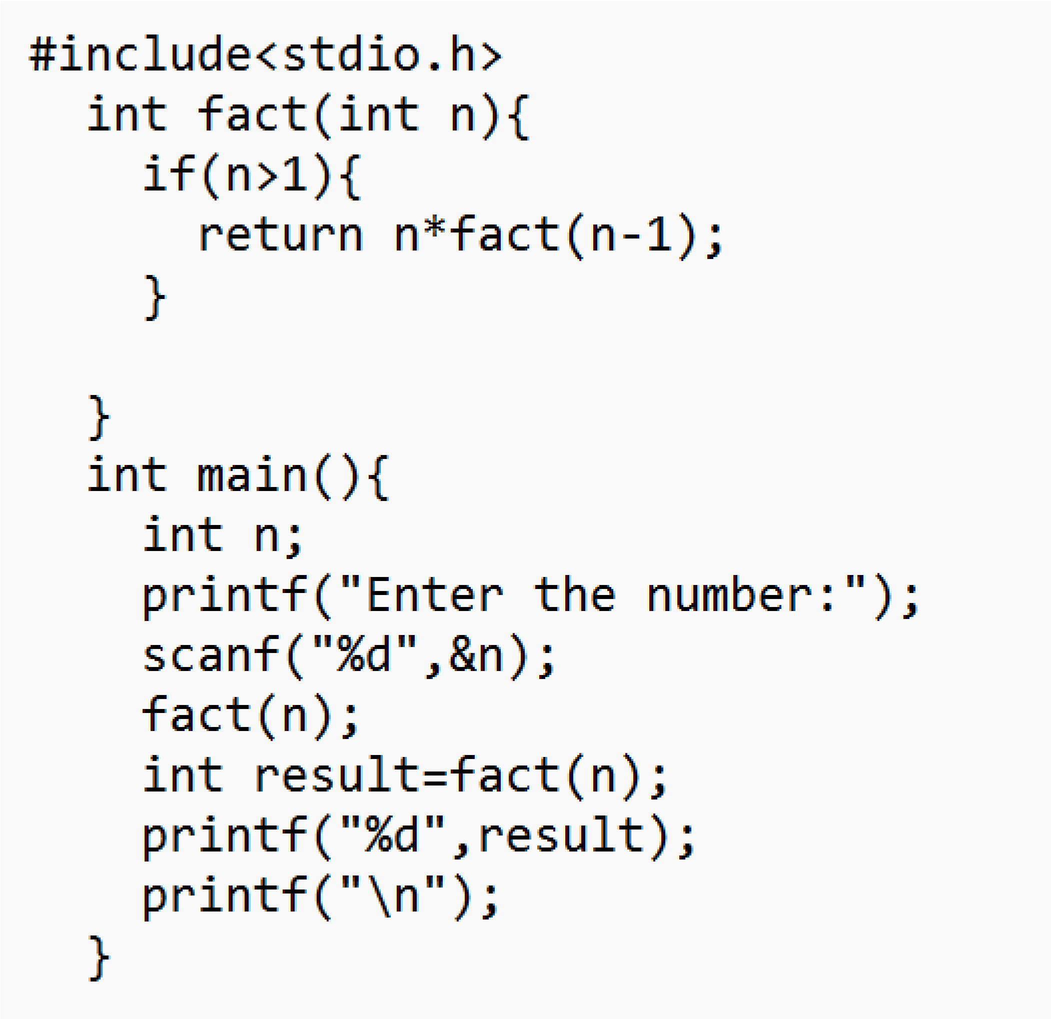


```
root@amma43:/home/amma/Documents# gcc -o cubes cubes.c
root@amma43:/home/amma/Documents# ./cubes
Enter the value of n:3
36
```

Space complexity: Same as above problem ,It uses only fixed size variable and it uses only one and it keeps changing for every iteration and no new variable is used so the space complexity for this code is **0(1)**

4)Problem statement: Write a program to write factorial of an given integer using recursion.

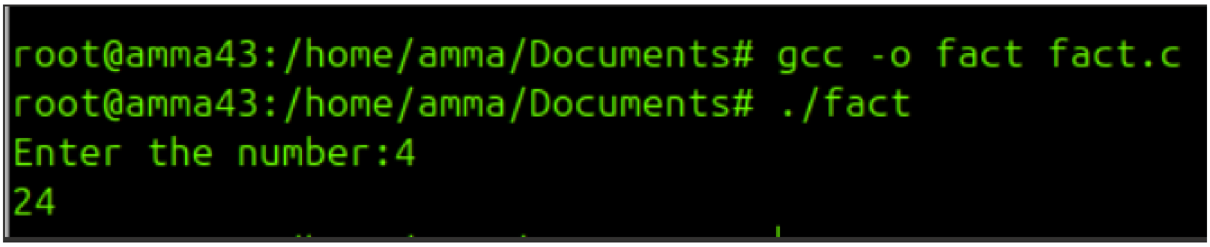
Code:



```
#include<stdio.h>
 int fact(int n){
  if(n>1){
   return n*fact(n-1);
  }

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

Output:

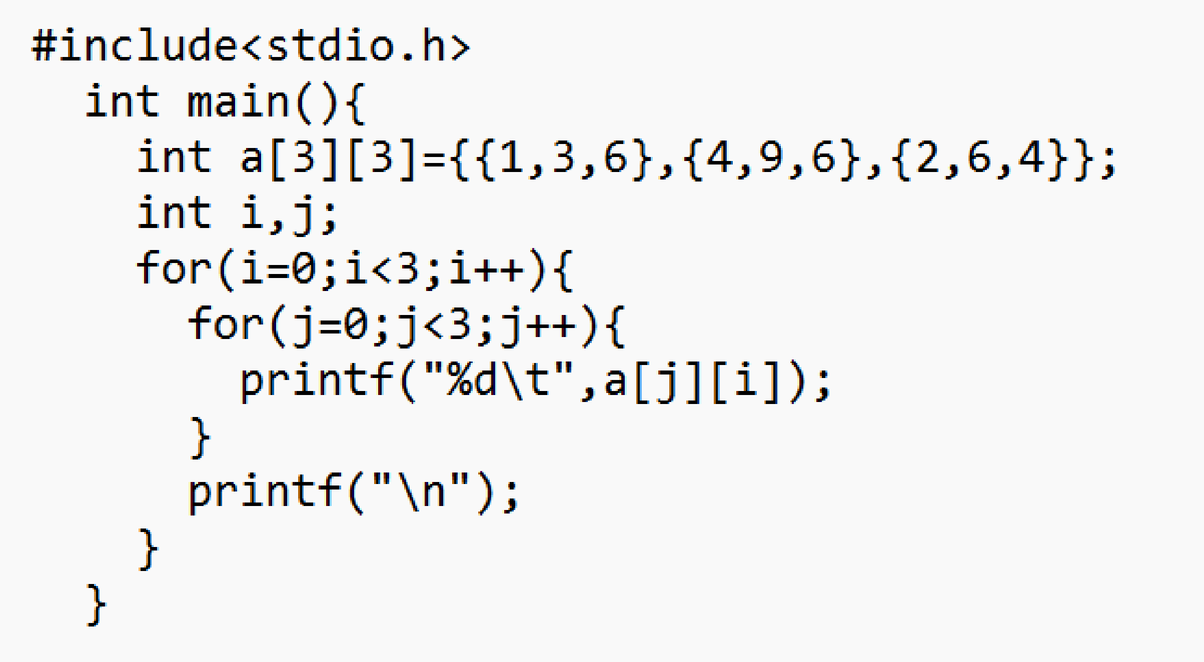


```
root@amma43:/home/amma/Documents# gcc -o fact fact.c
root@amma43:/home/amma/Documents# ./fact
Enter the number:4
24
```

Space complexity: In this program the function calls the variable for n times and each time it stores in the inner memory and hence it uses **O(n)** space complexity.

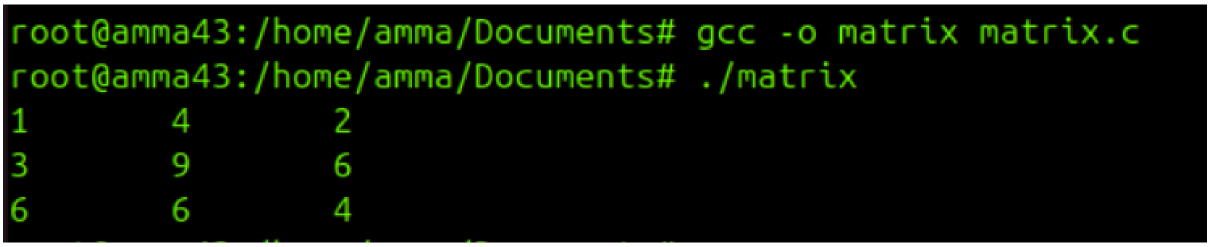
5)Problem statement: Write a program for transposing a 3x3 matrix.

Code:



```
#include<stdio.h>
  int main(){
    int a[3][3]={{1,3,6},{4,9,6},{2,6,4}};
    int i,j;
    for(i=0;i<3;i++){
      for(j=0;j<3;j++){
        printf("%d\t",a[j][i]);
      }
      printf("\n");
    }
  }
```

Output:

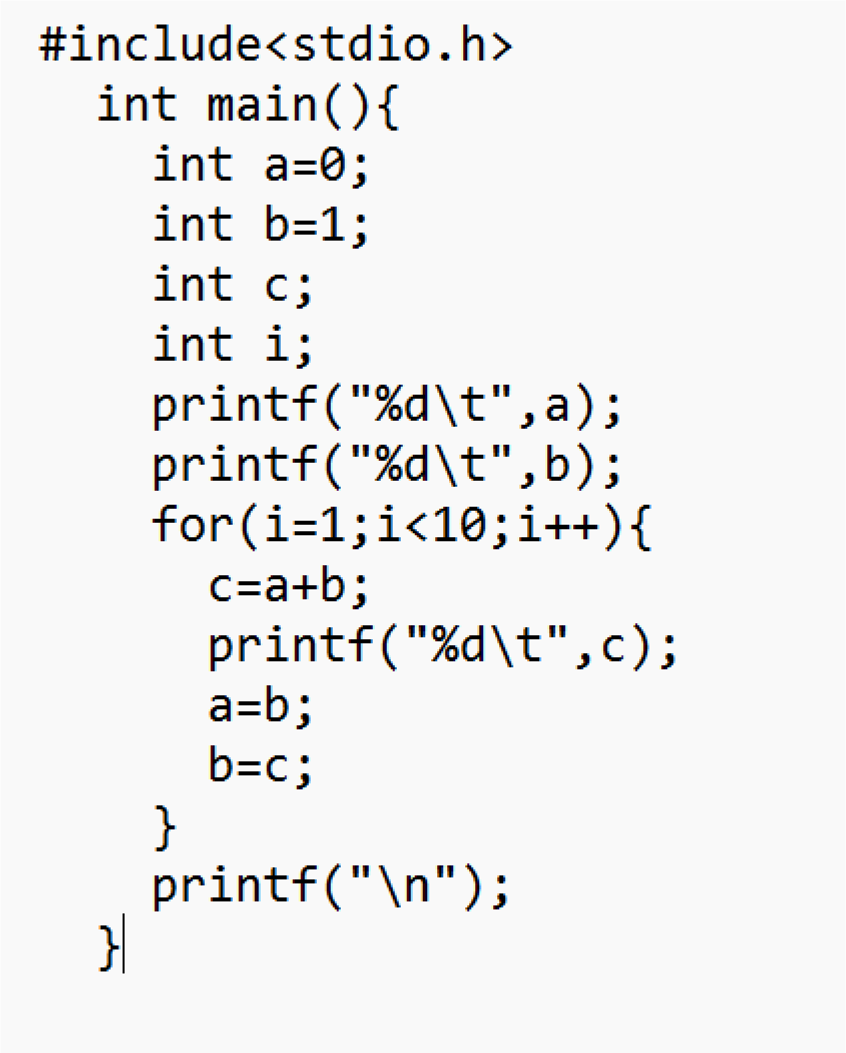


```
root@amma43:/home/amma/Documents# gcc -o matrix matrix.c
root@amma43:/home/amma/Documents# ./matrix
1       4       2
3       9       6
6       6       4
```

Space complexity: No additional memory is used and every variable is fixed variable so the space complexity will be **O(1)** as same as first three problems.

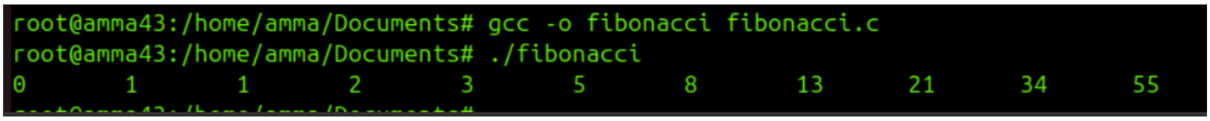
6)Problem statement: Write a program to find Fibonacci series.

Code:



```
#include<stdio.h>
  int main(){
    int a=0;
    int b=1;
    int c;
    int i;
    printf("%d\t",a);
    printf("%d\t",b);
    for(i=1;i<10;i++){
      c=a+b;
      printf("%d\t",c);
      a=b;
      b=c;
    }
    printf("\n");
  }
```

<u>Output:</u>



```
root@amma43:/home/amma/Documents# gcc -o fibonacci fibonacci.c
root@amma43:/home/amma/Documents# ./fibonacci
0       1       1       2       3       5       8       13      21      34      55
root@amma43:/home/amma/Documents#
```

Space complexity: Same as above problem no memory is used and all the variables are fixed variables,same variables keeps changing but no external variable is taken.so the time complexity is **O(1).**