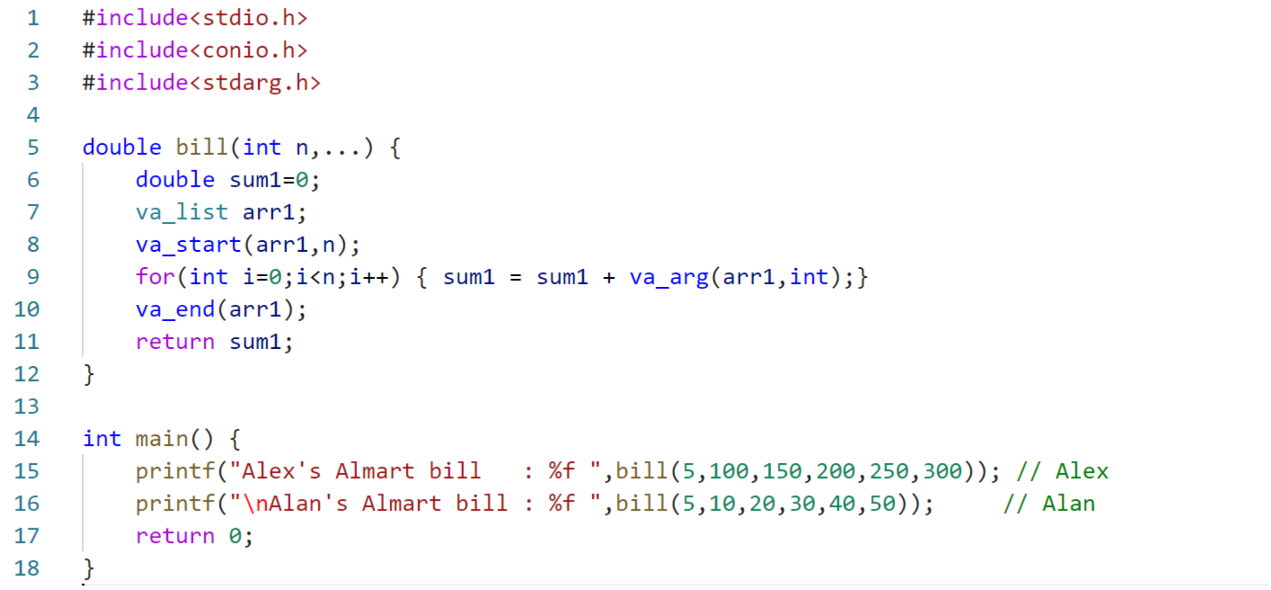
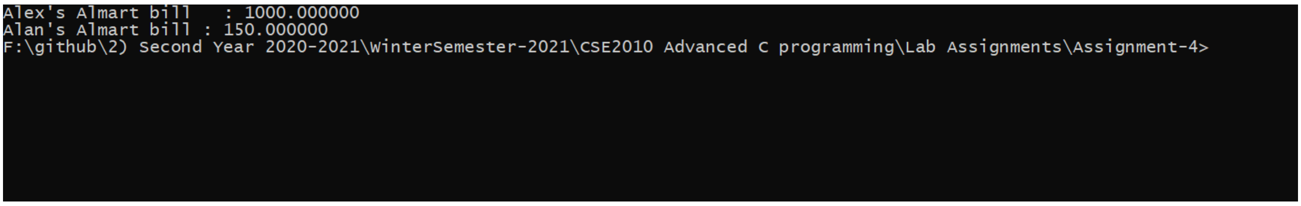
1. Alex and Alan bought n different products from Allmart. Write a C program to calculate separate bill for them, by passing prices of product to functions using variable length arguments.

Code



```c
#include<stdio.h>
#include<conio.h>
#include<stdarg.h>

double bill(int n,...) {
    double sum1=0;
    va_list arr1;
    va_start(arr1,n);
    for(int i=0;i<n;i++) { sum1 = sum1 + va_arg(arr1,int);}
    va_end(arr1);
    return sum1;
}

int main() {
    printf("Alex's Almart bill   : %f ",bill(5,100,150,200,250,300)); // Alex
    printf("\nAlan's Almart bill : %f ",bill(5,10,20,30,40,50));     // Alan
    return 0;
}
```

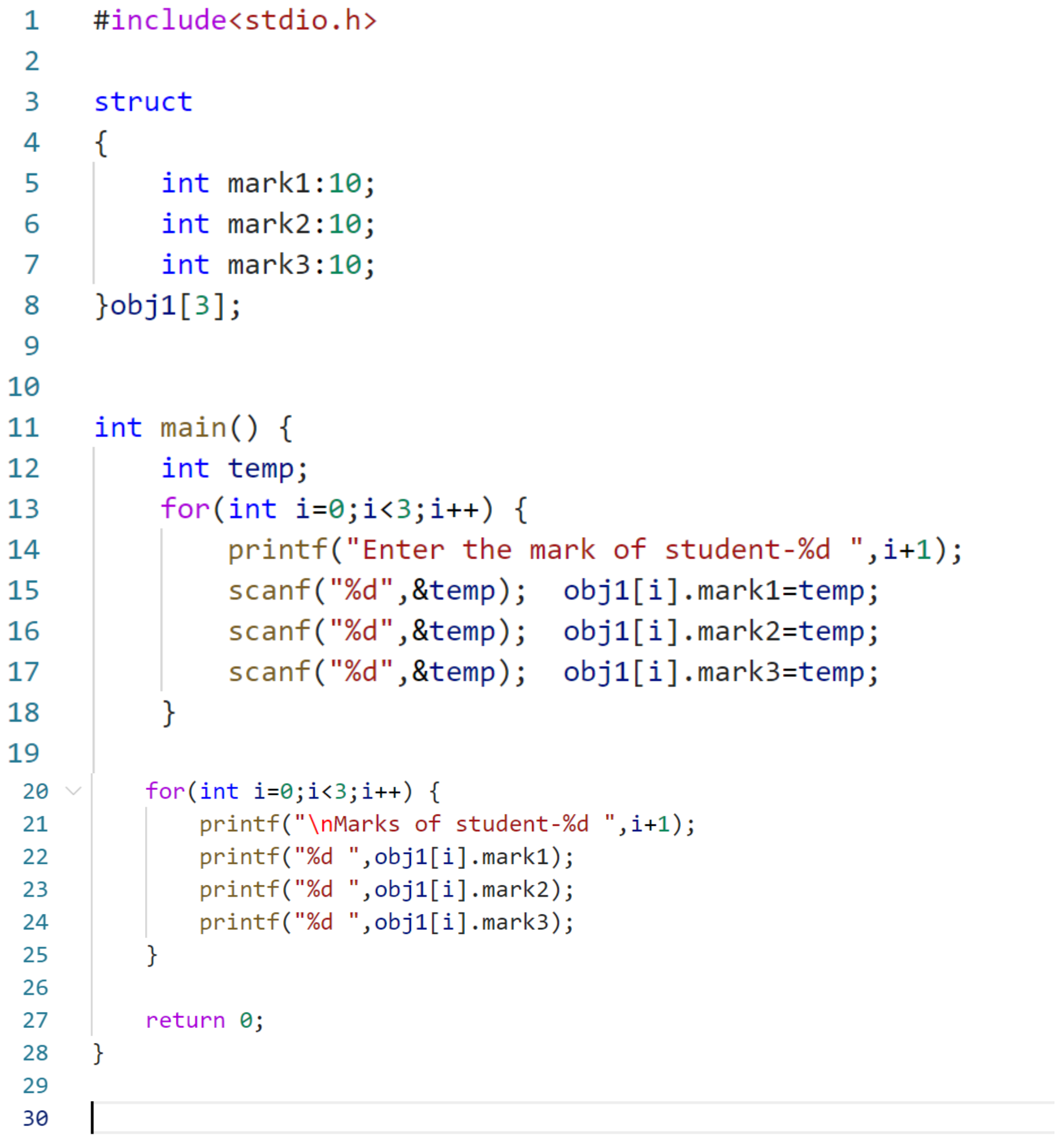
Output



```
Alex's Almart bill   : 1000.000000
Alan's Almart bill : 150.000000
F:\github\2) Second Year 2020-2021\WinterSemester-2021\CSE2010 Advanced C programming\Lab Assignments\Assignment-4>
```

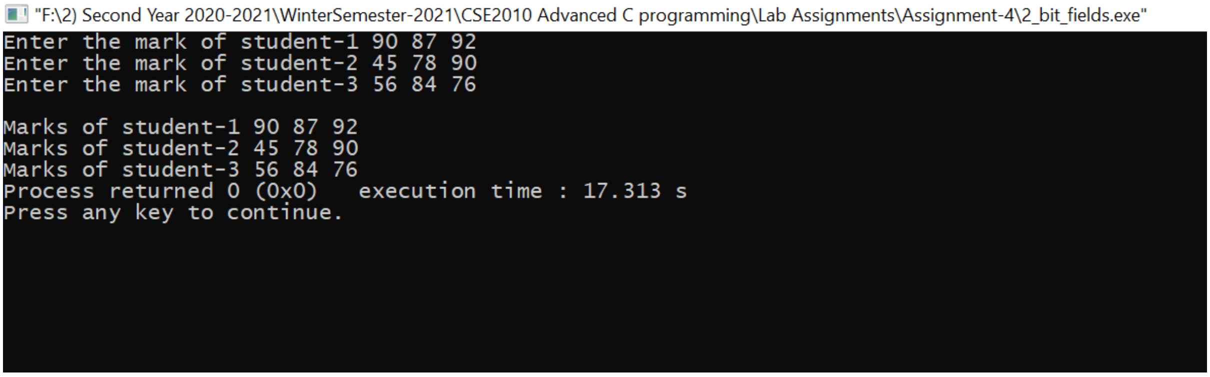
2. Write a C program to display the grade of n students by calculating the average of 3 marks using bit fields.

Code



```c
#include<stdio.h>

struct
{
    int mark1:10;
    int mark2:10;
    int mark3:10;
}obj1[3];


int main() {
    int temp;
    for(int i=0;i<3;i++) {
        printf("Enter the mark of student-%d ",i+1);
        scanf("%d",&temp);  obj1[i].mark1=temp;
        scanf("%d",&temp);  obj1[i].mark2=temp;
        scanf("%d",&temp);  obj1[i].mark3=temp;
    }

    for(int i=0;i<3;i++) {
        printf("\nMarks of student-%d ",i+1);
        printf("%d ",obj1[i].mark1);
        printf("%d ",obj1[i].mark2);
        printf("%d ",obj1[i].mark3);
    }

    return 0;
}

```

Output

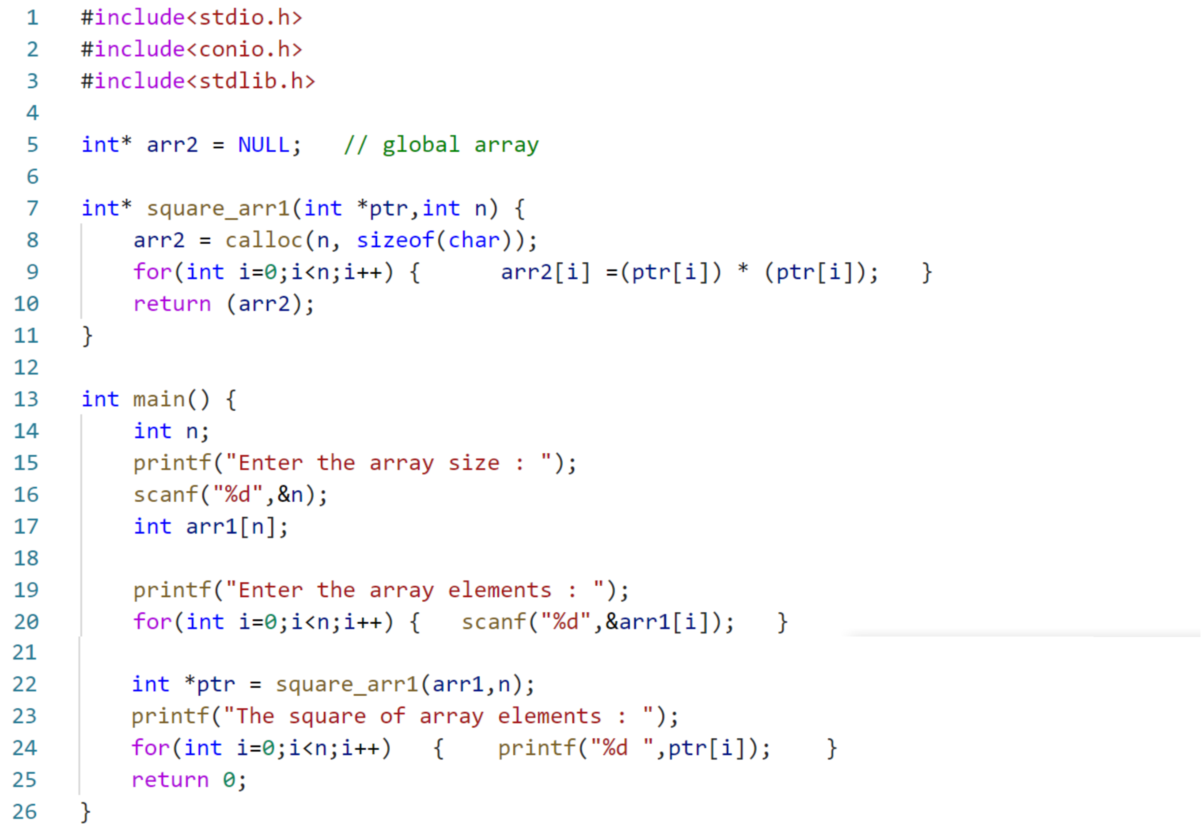


"F:\2) Second Year 2020-2021\WinterSemester-2021\CSE2010 Advanced C programming\Lab Assignments\Assignment-4\2_bit_fields.exe"

```
Enter the mark of student-1 90 87 92
Enter the mark of student-2 45 78 90
Enter the mark of student-3 56 84 76

Marks of student-1 90 87 92
Marks of student-2 45 78 90
Marks of student-3 56 84 76
Process returned 0 (0x0)   execution time : 17.313 s
Press any key to continue.
```

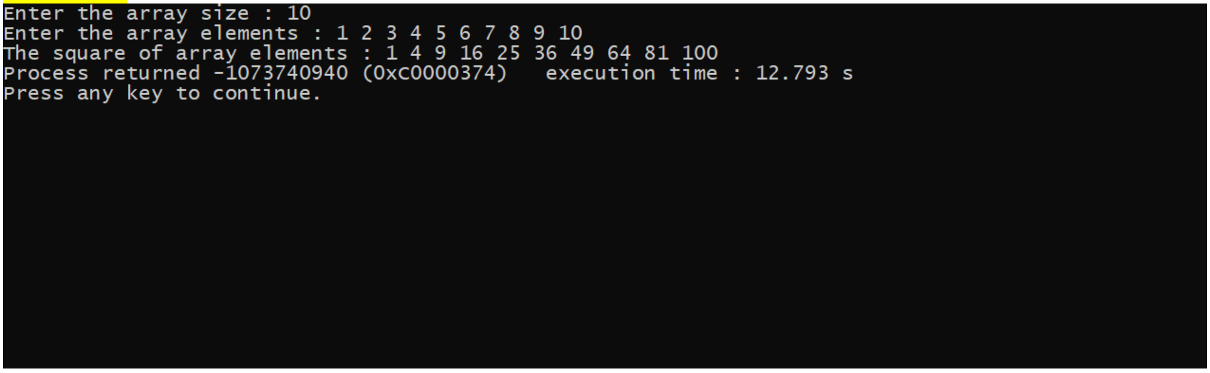
3. Write a C program to create an array A of n numbers. Create another array B with the square of elements of array A, by using passing and returning array as pointers to function.

Code



```c
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>

int* arr2 = NULL;   // global array

int* square_arr1(int *ptr,int n) {
    arr2 = calloc(n, sizeof(char));
    for(int i=0;i<n;i++) {      arr2[i] =(ptr[i]) * (ptr[i]);    }
    return (arr2);
}

int main() {
    int n;
    printf("Enter the array size : ");
    scanf("%d",&n);
    int arr1[n];

    printf("Enter the array elements : ");
    for(int i=0;i<n;i++) {   scanf("%d",&arr1[i]);    }

    int *ptr = square_arr1(arr1,n);
    printf("The square of array elements : ");
    for(int i=0;i<n;i++)   {    printf("%d ",ptr[i]);     }
    return 0;
}
```

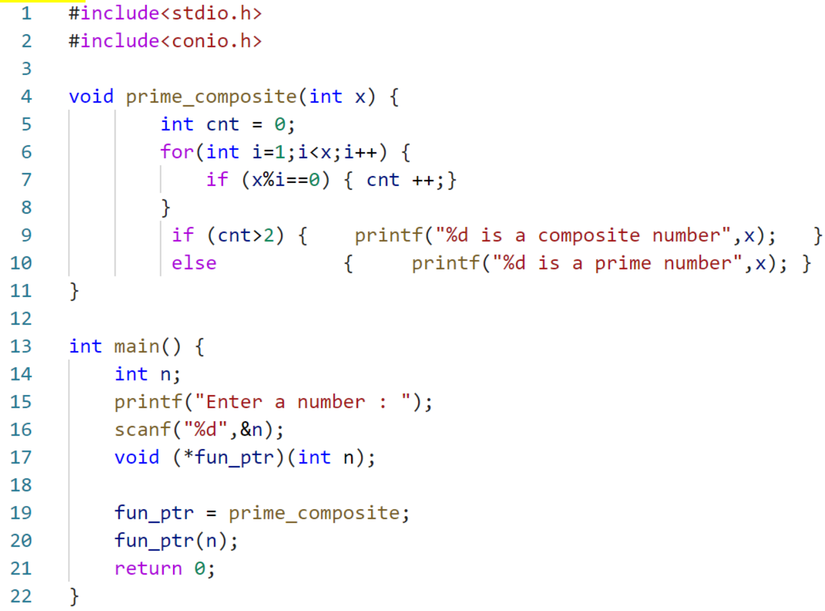
Output



```
Enter the array size : 10
Enter the array elements : 1 2 3 4 5 6 7 8 9 10
The square of array elements : 1 4 9 16 25 36 49 64 81 100
Process returned -1073740940 (0xC0000374)   execution time : 12.793 s
Press any key to continue.
```

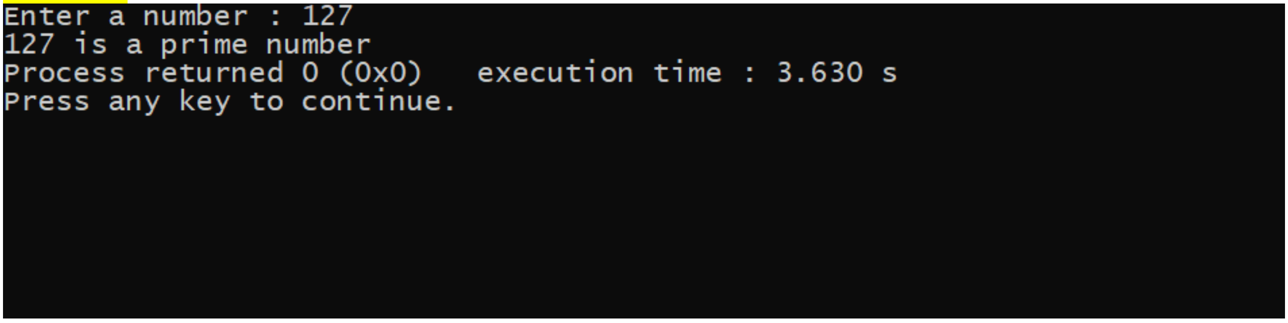
4. Write a C program to find whether a given number is prime or not using function pointer.

Code



```c
#include<stdio.h>
#include<conio.h>

void prime_composite(int x) {
        int cnt = 0;
        for(int i=1;i<x;i++) {
            if (x%i==0) { cnt ++;}
        }
         if (cnt>2) {    printf("%d is a composite number",x);   }
         else           {     printf("%d is a prime number",x); }
}

int main() {
    int n;
    printf("Enter a number : ");
    scanf("%d",&n);
    void (*fun_ptr)(int n);

    fun_ptr = prime_composite;
    fun_ptr(n);
    return 0;
}
```

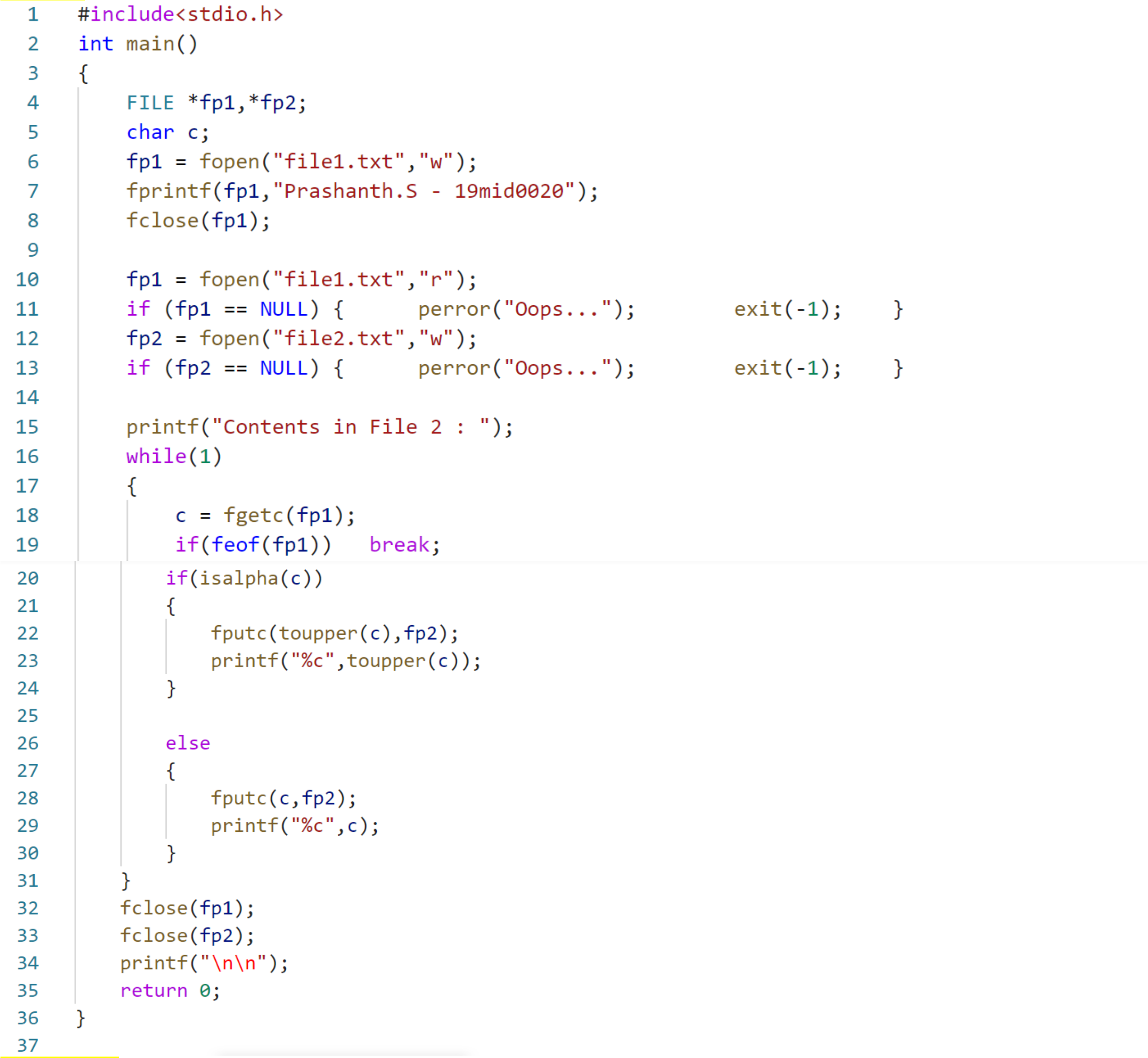
Output



```
Enter a number : 127
127 is a prime number
Process returned 0 (0x0)   execution time : 3.630 s
Press any key to continue.
```

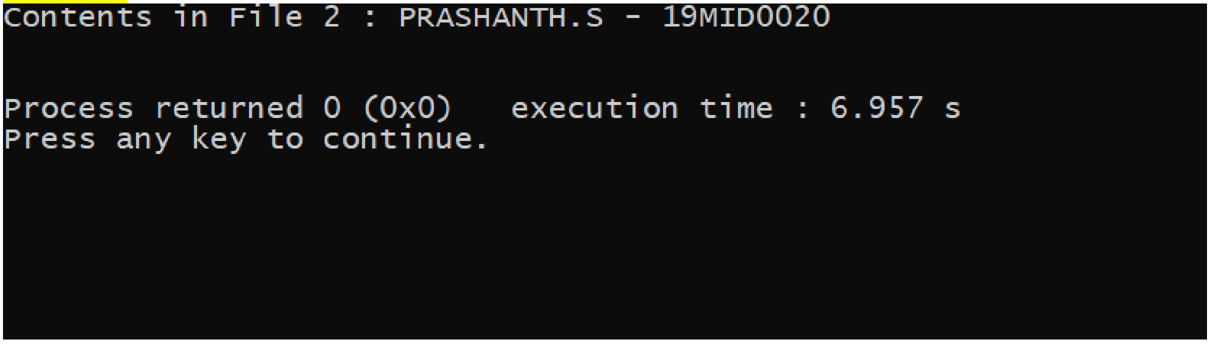
5. Write a C program to create a file F1 with a string1. Create another file F2 from F1 such that all the characters of string1 as upper case and print the result in screen. While creating the files check for the validity of file creation.

Code



```c
#include<stdio.h>
int main()
{
    FILE *fp1,*fp2;
    char c;
    fp1 = fopen("file1.txt","w");
    fprintf(fp1,"Prashanth.S - 19mid0020");
    fclose(fp1);

    fp1 = fopen("file1.txt","r");
    if (fp1 == NULL) {      perror("Oops...");        exit(-1);    }
    fp2 = fopen("file2.txt","w");
    if (fp2 == NULL) {      perror("Oops...");        exit(-1);    }

    printf("Contents in File 2 : ");
    while(1)
    {
        c = fgetc(fp1);
        if(feof(fp1))   break;
        if(isalpha(c))
        {
            fputc(toupper(c),fp2);
            printf("%c",toupper(c));
        }

        else
        {
            fputc(c,fp2);
            printf("%c",c);
        }
    }
    fclose(fp1);
    fclose(fp2);
    printf("\n\n");
    return 0;
}

```

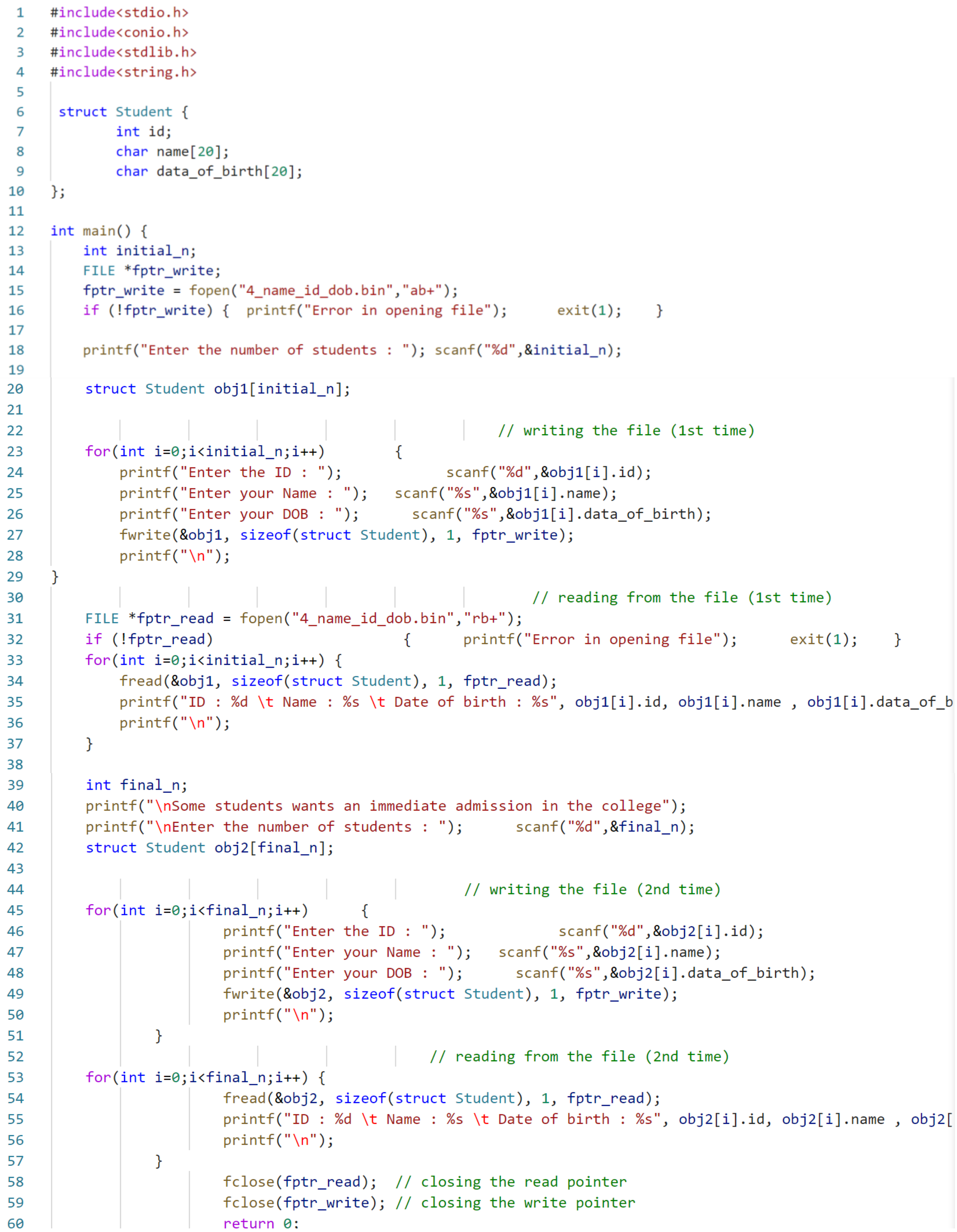
Output



```
Contents in File 2 : PRASHANTH.S - 19MID0020


Process returned 0 (0x0)   execution time : 6.957 s
Press any key to continue.
```

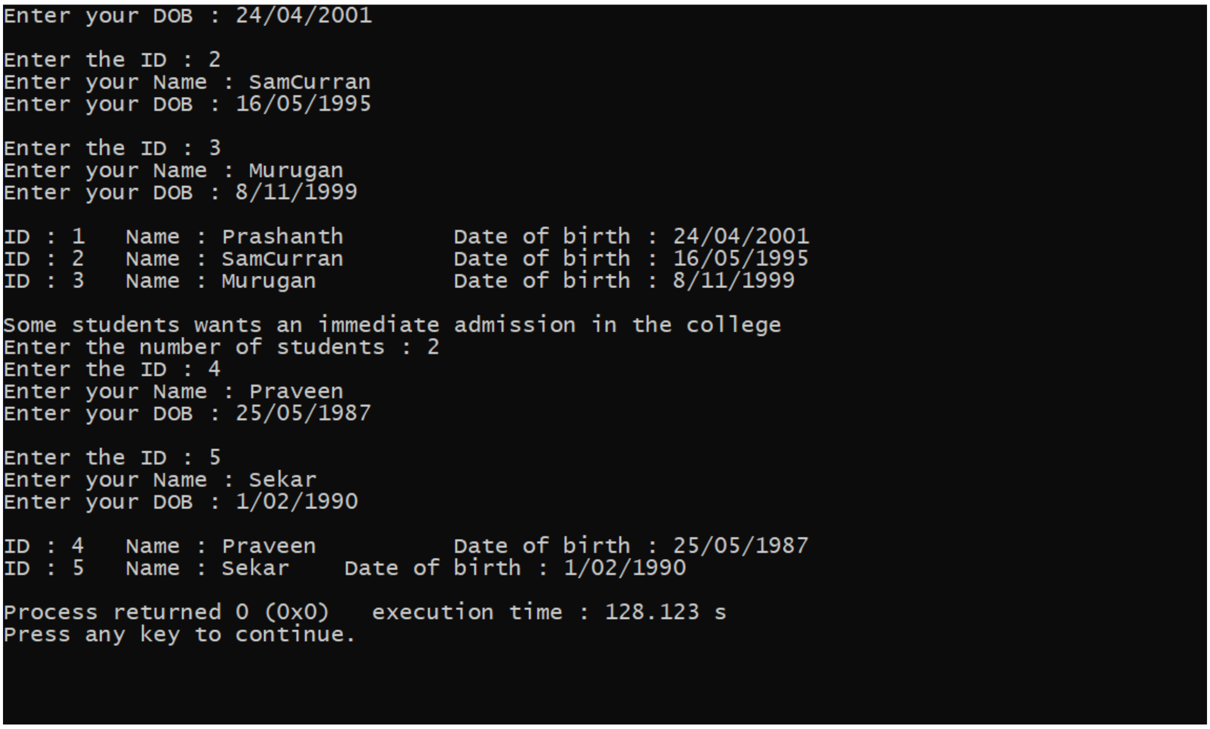
6. Write a C program to create a student record with **name, id** and **date of birth** and store it as a binary file. Later append another record of student to the same file.

Code



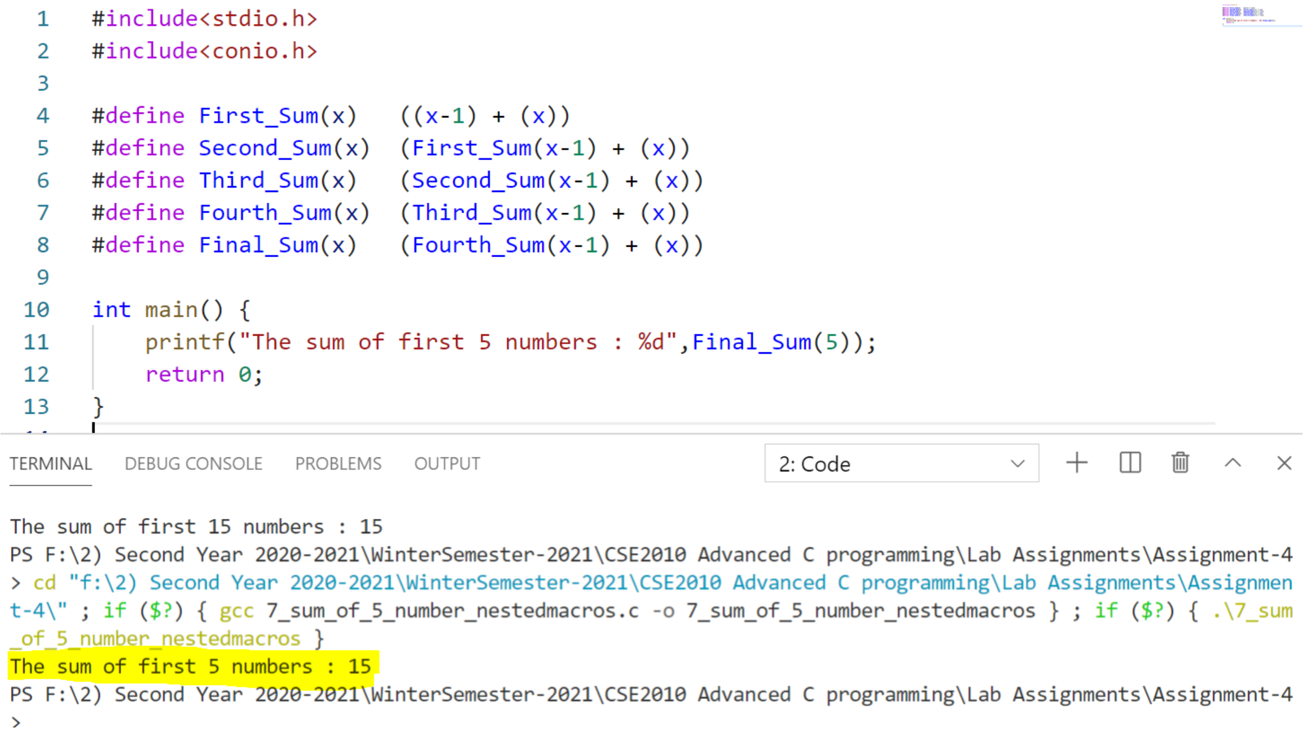
```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<string.h>

 struct Student {
        int id;
        char name[20];
        char data_of_birth[20];
};

int main() {
    int initial_n;
    FILE *fptr_write;
    fptr_write = fopen("4_name_id_dob.bin","ab+");
    if (!fptr_write) {  printf("Error in opening file");      exit(1);    }

    printf("Enter the number of students : "); scanf("%d",&initial_n);

    struct Student obj1[initial_n];

                                                      // writing the file (1st time)
    for(int i=0;i<initial_n;i++)        {
        printf("Enter the ID : ");              scanf("%d",&obj1[i].id);
        printf("Enter your Name : ");   scanf("%s",&obj1[i].name);
        printf("Enter your DOB : ");      scanf("%s",&obj1[i].data_of_birth);
        fwrite(&obj1, sizeof(struct Student), 1, fptr_write);
        printf("\n");
}
                                                          // reading from the file (1st time)
    FILE *fptr_read = fopen("4_name_id_dob.bin","rb+");
    if (!fptr_read)                     {      printf("Error in opening file");      exit(1);    }
    for(int i=0;i<initial_n;i++) {
        fread(&obj1, sizeof(struct Student), 1, fptr_read);
        printf("ID : %d \t Name : %s \t Date of birth : %s", obj1[i].id, obj1[i].name , obj1[i].data_of_b
        printf("\n");
    }

    int final_n;
    printf("\nSome students wants an immediate admission in the college");
    printf("\nEnter the number of students : ");      scanf("%d",&final_n);
    struct Student obj2[final_n];

                                                  // writing the file (2nd time)
    for(int i=0;i<final_n;i++)      {
                    printf("Enter the ID : ");              scanf("%d",&obj2[i].id);
                    printf("Enter your Name : ");   scanf("%s",&obj2[i].name);
                    printf("Enter your DOB : ");      scanf("%s",&obj2[i].data_of_birth);
                    fwrite(&obj2, sizeof(struct Student), 1, fptr_write);
                    printf("\n");
            }
                                              // reading from the file (2nd time)
    for(int i=0;i<final_n;i++) {
                    fread(&obj2, sizeof(struct Student), 1, fptr_read);
                    printf("ID : %d \t Name : %s \t Date of birth : %s", obj2[i].id, obj2[i].name , obj2[
                    printf("\n");
            }
                    fclose(fptr_read);  // closing the read pointer
                    fclose(fptr_write); // closing the write pointer
                    return 0;
```

Output



```
Enter your DOB : 24/04/2001

Enter the ID : 2
Enter your Name : SamCurran
Enter your DOB : 16/05/1995

Enter the ID : 3
Enter your Name : Murugan
Enter your DOB : 8/11/1999

ID : 1   Name : Prashanth         Date of birth : 24/04/2001
ID : 2   Name : SamCurran         Date of birth : 16/05/1995
ID : 3   Name : Murugan           Date of birth : 8/11/1999

Some students wants an immediate admission in the college
Enter the number of students : 2
Enter the ID : 4
Enter your Name : Praveen
Enter your DOB : 25/05/1987

Enter the ID : 5
Enter your Name : Sekar
Enter your DOB : 1/02/1990

ID : 4   Name : Praveen           Date of birth : 25/05/1987
ID : 5   Name : Sekar     Date of birth : 1/02/1990

Process returned 0 (0x0)   execution time : 128.123 s
Press any key to continue.
```

7. Write a C program to find the sum of 5 numbers using nested macro.

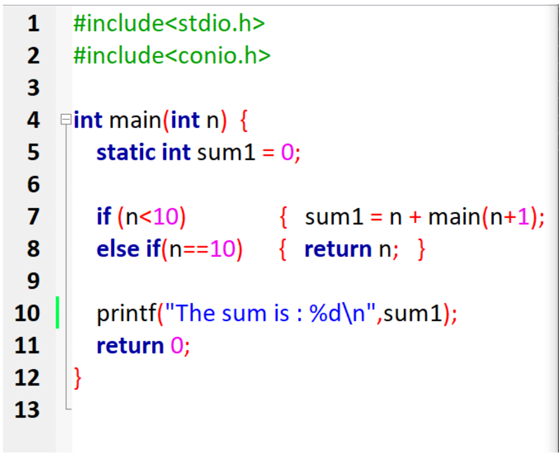


```c
#include<stdio.h>
#include<conio.h>

#define First_Sum(x)   ((x-1) + (x))
#define Second_Sum(x)  (First_Sum(x-1) + (x))
#define Third_Sum(x)   (Second_Sum(x-1) + (x))
#define Fourth_Sum(x)  (Third_Sum(x-1) + (x))
#define Final_Sum(x)   (Fourth_Sum(x-1) + (x))

int main() {
    printf("The sum of first 5 numbers : %d",Final_Sum(5));
    return 0;
}
```

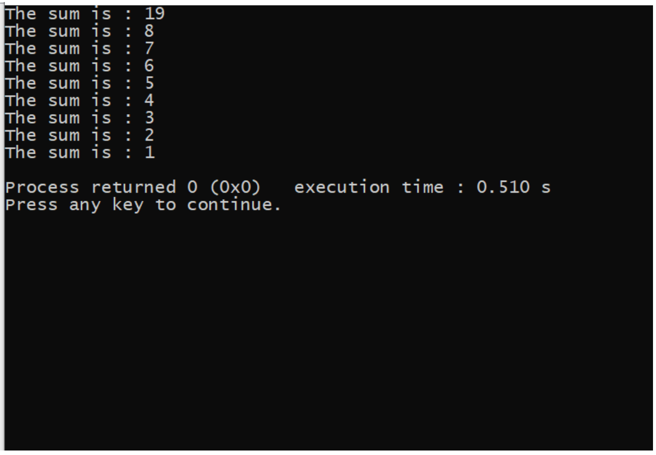
```
The sum of first 15 numbers : 15
PS F:\2) Second Year 2020-2021\WinterSemester-2021\CSE2010 Advanced C programming\Lab Assignments\Assignment-4
> cd "f:\2) Second Year 2020-2021\WinterSemester-2021\CSE2010 Advanced C programming\Lab Assignments\Assignmen
t-4\" ; if ($?) { gcc 7_sum_of_5_number_nestedmacros.c -o 7_sum_of_5_number_nestedmacros } ; if ($?) { .\7_sum
_of_5_number_nestedmacros }
The sum of first 5 numbers : 15
PS F:\2) Second Year 2020-2021\WinterSemester-2021\CSE2010 Advanced C programming\Lab Assignments\Assignment-4
>
```

8. Just for learning: Write a C program to call main() function recursively and perform sum of 1 to 10 numbers.

Code



```c
#include<stdio.h>
#include<conio.h>

int main(int n) {
    static int sum1 = 0;

    if (n<10)           {  sum1 = n + main(n+1);
    else if(n==10)      {  return n;  }

    printf("The sum is : %d\n",sum1);
    return 0;
}
```



```
The sum is : 19
The sum is : 8
The sum is : 7
The sum is : 6
The sum is : 5
The sum is : 4
The sum is : 3
The sum is : 2
The sum is : 1

Process returned 0 (0x0)   execution time : 0.510 s
Press any key to continue.
```