Raunak Gayen
2020EEB066

1. Area of a triangle is given by the formula $A = \sqrt{S(S-a)\ (S-b)\ (S-c)}$ where a, b, and c are the sides of the triangle and $2S = a + b + c$. Write a C program to compute the area of the triangle by taking the values of *a*, *b*, and *c* as CONSTANTS.

```c
#include<stdio.h>
#include<math.h>
int main()
{
    float a=3.0, b=4.0, c=5.0;
    float area, S;
    S=(a+b+c)/2;
    area = sqrt(S*(S-a)*(S-b)*(S-c));
    printf("%f\n",area);
    return 0;
}

```

Output:

```
Microsoft Windows [Version 10.0.19042.685]
(c) 2020 Microsoft Corporation. All rights reserved.

D:\compAssignment>gcc triangle.c

D:\compAssignment>a.exe
6.000000
```

2. Write a C program that reads three integer values from the keyboard and determines if they are the sides of a right-angled triangle.

Source Code:

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

int main()
{
    int a,b,c;
    scanf("%d%d%d",&a,&b,&c);
    if(a>b && a>c)// a is hypo
    {
        if(b*b + c*c == a*a)
            printf("Right-angled Triangle\n");
        else
            printf("Not a Right-angled Triangle\n");
    }
    else if(b>a && b>c)// b is hypo
    {
        if(a*a + c*c == b*b)
            printf("Right-angled Triangle\n");
        else
            printf("Not a Right-angled Triangle\n");
    }
    else if(c>a && c>b)// c is hypo
    {
        if(b*b + a*a == c*c)
            printf("Right-angled Triangle\n");
        else
            printf("Not a Right-angled Triangle\n");
    }
    else
        printf("Not a Right-angled triangle\n");
    return 0;

}
```

Output:

```
Microsoft Windows [Version 10.0.19042.685]
(c) 2020 Microsoft Corporation. All rights reserved.

D:\compAssignment>gcc rtangle.c

D:\compAssignment>a.exe
3
4
5
Right-angled Triangle
```

```
D:\compAssignment>a.exe
3 4 6
Not a Right-angled Triangle
```

3. Write a C program that prints the next character from the ASCII table for the corresponding character taken from the keyboard. As for example if the character taken from the keyboard is 'B' the program will print the immediate next character 'C'. Also print the corresponding ASCII value.

```c
#include<stdio.h>
int main()
{
    char c,nc;
    printf("Enter a character : \n");
    scanf("%c", &c );
    nc=c+1;
    printf("The required character and integer : %c and %d",nc,nc);
    return 0;
}
```

Output:

```
D:\compAssignment>gcc ascii.c

D:\compAssignment>a.exe
Enter a character :
A
The required character and integer : B and 66
```

4. Write a C program to find the sum of individual digits of a positive integer. The number has to be taken as input from the keyboard. Keep a check to accept only the numbers with maximum of three digits.

Source Code:

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

int main()
{
    int num;
    printf("Enter a 3 digit number\n");
    scanf("%d",&num);
    if(num>=100 && num<=999)
    {
        int sum=0;
        while(num!=0)
        {
            sum=sum+ (num%10);
            num=num/10;
        }
        printf("Sum=%d",sum);
    }
    else
        printf("Invalid Input\n");
    return 0;
}
```

Output:

```
Microsoft Windows [Version 10.0.19042.685]
(c) 2020 Microsoft Corporation. All rights reserved.

D:\compAssignment>gcc sumofdigits.c

D:\compAssignment>a.exe
Enter a 3 digit number
426
Sum=12
D:\compAssignment>a.exe
Enter a 3 digit number
9999
Invalid Input
```

5. Write a C program to test for odd or even of an integer by the following two methods: (i) division and (ii) bit testing. The integer has to be taken as input from the console.

```c
#include<stdio.h>
#include<math.h>
int main()
{
    int n,ch;//int bin
    printf("Enter your choice '1' or '2' and the number consecutively :");
    scanf("%d %d", &ch , &n );
    switch (ch)
    {
    case 1:
        if( ( 2 * ( n / 2 ) ) == n )
            printf("%d is even.\n",n);
        else
            printf("%d is odd.\n",n);
        break;
    case 2:
        if(n&1)
            printf("%d is odd.\n",n);
        else
            printf("%d is even.\n",n);

        break;
    default:
        printf("Invalid Input");
        break;
    }
    return 0;
}
```

Output:

```
D:\compAssignment>gcc oddeven.c

D:\compAssignment>a.exe
Enter your choice '1' or '2' and the number consecutively :
1 24
24 is even.

D:\compAssignment>a.exe
Enter your choice '1' or '2' and the number consecutively :
2 19
19 is odd.
```

6. Using Switch-case block, input the name of the month from the user and display the corresponding number of days in that month.

```c
#include<stdio.h>
int main()
{
    int n;
    printf("Enter month number\n");
    scanf("%d",&n);
    switch (n)
    {
        case 1:
            printf("January - 31 days\n");
            break;
        case 2:
            printf("February - 28 days\n");
            break;
        case 3:
            printf("March - 31 days\n");
            break;
        case 4:
            printf("April - 30 days\n");
            break;
        case 5:
            printf("May - 30 days\n");
            break;
        case 6:
            printf("June - 30 days\n");
            break;
        case 7:
            printf("July - 31 days\n");
            break;
        case 8:
            printf("August - 31 days\n");
            break;
        case 9:
            printf("September - 30 days\n");
            break;
        case 10:
            printf("October - 31 days\n");
            break;
        case 11:
            printf("November - 30 days\n");
            break;
        case 12:
            printf("December - 31 days\n");
            break;
        default:
            printf("Invalid Input");
            break;
    }
    return 0;
}
```

Output:

```
D:\compAssignment>gcc month.c

D:\compAssignment>a.exe
Enter month number
2
February - 28 days
```