

Birla Institute of Technology & Science, Pilani, Hyderabad Campus

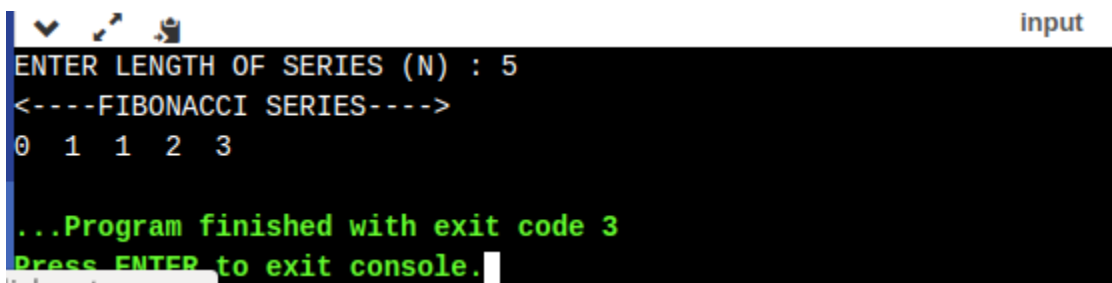
Second Semester 2020-2021

Computer Programming [CS F111]

Lab 4

Q1. Print the Fibonacci series for a given n.

```
1  #include <stdio.h>
2
3  void main()
4  {
5      int num1=0, num2=1,no,counter,fab;
6
7      printf("ENTER LENGTH OF SERIES (N) : ");
8      scanf("%d",&no);
9
10     printf("<----FIBONACCI SERIES---->");
11     printf("\n%d  %d",num1,num2);
12     for(counter = 1; counter <= no-2; counter++)
13     {
14         fab=num1 + num2;
15         printf("  %d",fab);
16         num1=num2;
17         num2=fab;
18     }
19 }
20
```



input

```
ENTER LENGTH OF SERIES (N) : 5
<----FIBONACCI SERIES---->
0 1 1 2 3
...Program finished with exit code 3
Press ENTER to exit console.
```

Q2. Calculate the square root of a number n without using predefined functions (library functions).

```

1  #include<stdio.h>
2
3  void main()
4  {
5      int n;
6
7      float temp, squareroor;
8
9      printf("Provide the number: ");
10
11     scanf("%d", &n);
12     squareroor = n / 2;
13     temp = 0;
14
15     while(squareroor != temp){
16         temp = squareroor;
17         squareroor = ( n/temp + temp) / 2;
18     }
19
20     printf("The square root of '%d' is '%f'", n, squareroor);
21 }
22

```

input

```

Provide the number: 36
The square root of '36' is '6.000000'

...Program finished with exit code 37
Press ENTER to exit console.

```

Q3. Print all prime numbers between 0 and n. Take n as input.

```

1  #include <stdio.h>
2
3  void main()
4  {
5      int no,counter,counter1,check;
6      printf("INPUT THE VALUE OF N: ");
7      scanf("%d",&no);
8      printf("\nTHE PRIME NO. SERIES B/W 1 TO %d : \n",no);
9
10     for(counter = 2; counter <= no; counter++)
11     {
12         check = 0;
13         for(counter1 = counter-1; counter1 > 1 ; counter1--)
14             if(counter%counter1 == 0)
15             {
16                 check++;
17                 break;
18             }
19         if(check == 0)
20             printf("%d ",counter);
21     }
22 }

```

```
input
INPUT THE VALUE OF N: 20

THE PRIME NO. SERIES B/W 1 TO 20 :
2 3 5 7 11 13 17 19

...Program finished with exit code 20
```

Q4. Write a program to calculate the sum of the digits of an integer.

```
1  #include <stdio.h>
2
3  void main()
4  {
5      int num, k = 1, sum = 0;
6      printf("Enter the number whose digits are to be added:");
7      scanf("%d", &num);
8      while (num != 0)
9      {
10         k = num % 10;
11         sum = sum + k;
12         k = num / 10;
13         num = k;
14     }
15     printf("Sum of the digits:%d", sum);
16 }
17
```

```
input
Enter the number whose digits are to be added:5461
Sum of the digits:16

...Program finished with exit code 20
Press ENTER to exit console.
```

Q5. Check whether the given number is Armstrong or not.

```

1  #include <math.h>
2  #include <stdio.h>
3
4  int main() {
5      int num, originalNum, remainder, n = 0;
6      float result = 0.0;
7
8      printf("Enter an integer: ");
9      scanf("%d", &num);
10
11     originalNum = num;
12
13     // store the number of digits of num in n
14     for (originalNum = num; originalNum != 0; ++n) {
15         originalNum /= 10;
16     }
17
18     for (originalNum = num; originalNum != 0; originalNum /= 10) {
19         remainder = originalNum % 10;
20
21         // store the sum of the power of individual digits in result
22         result += pow(remainder, n);
23     }
24
25     // if num is equal to result, the number is an Armstrong number
26     if ((int)result == num)
27         printf("%d is an Armstrong number.", num);
28     else
29         printf("%d is not an Armstrong number.", num);
30     return 0;
31 }
32

```

```

input
Enter an integer: 371
371 is an Armstrong number.

...Program finished with exit code 0
Press ENTER to exit console.

```

Q6. Write a C program to print taxicab numbers.

Hint: Taxicab, also called the n th Hardy–Ramanujan number, is defined as the smallest integer that can be expressed as a sum of two positive integer cubes in n distinct ways. The most famous taxicab number is $1729 = Ta = 1^3 + 12^3 = 9^3 + 10^3$.

```

9  #include <math.h>
10 #include <stdio.h>
11
12 int main()
13 {
14     int n,N,count=0;
15     unsigned long i=1;
16     printf("***** Print Taxicab numbers *****\n\n");
17     printf("Enter the number of distinct ways (keep it <= 2): ");
18     scanf("%d",&n);
19     printf("Enter the number of taxicab numbers you want to print: ");
20     scanf("%d",&N);
21
22     while(count<N){
23         int int_count = 0;
24
25         // Try all possible pairs (j, k) whose cube
26         // sums can be i.
27         for (unsigned long j = 1; j <= pow(i, 1.0/3); j++)
28             for (unsigned long k = j + 1; k <= pow(i, 1.0/3); k++)
29                 if (j*j*j + k*k*k == i)
30                     int_count++;
31
32         // Taxicab(2) found
33         if (int_count == n)
34         {
35             count++;
36             printf("count %d number %lu\n",count,i);
37         }
38
39         i++;
40     }
41
42     return 0;
43 }

```

```

***** Print Taxicab numbers *****

Enter the number of distinct ways (keep it <= 2): 2
Enter the number of taxicab numbers you want to print: 2
count 1 number 1729
count 2 number 4104

...Program finished with exit code 0
Press ENTER to exit console.

```

Assignment Questions

Q1. Write a C program to print Pascal's triangle using for loops by taking the limit from the user.

Hint: Pascal's triangle is a triangular array of the binomial coefficients.

The first thing one needs to know about Pascal's triangle is that all the numbers outside the triangle are "0"s. To build the triangle, start with a "1" at the top, then continue putting numbers below in a triangular pattern so as to form a triangular array. So, each new number added below the top "1" is just the sum of the two numbers above, except for the edge which are all "1"s. This can be summarized as:

0 row = 1

1 row = (0+1), (1+0) = 1, 1

2 row = (0+1), (1+1), (1+0) = 1, 2, 1

3 row = (0+1), (1+2), (2+1), (1+0) = 1, 3, 3, 1

4 row = (0+1), (1+3), (3+3), (3+1), (1+0) = 1, 4, 6, 4, 1

Q2. Write a C program to find the reverse of the number taken from the user.

Q3. Write a C program that accepts input from the user and checks if the number is equal to the sum of its positive divisors, excluding the number itself i.e. is a perfect number or not.

Q4. Write a C program to check if the given number is in binary format or not.

Hint: Here we are not going to check any base of the number. As you know we can represent any number in binary format like in 0 and 1. Just like the binary representation of 2 is 0010. So we will check only if the given input number has 0 and 1 or any other digits.

Q5. Write a C program to show the numbers, whose cubes are summed, to get the certain taxicab number.