

Jadavpur University
Session 2018-19, Odd Semester
Computer Programming and Numerical Methods
Sessional Assignments (Combined)

Flow Chart and Algorithms

1. Write an algorithm to determine maximum of three numbers. Also draw the corresponding flow chart.
2. Write an algorithm to determine the sum of individual digits of a given integer. Also draw the corresponding flow chart.
3. Write an algorithm to print the reverse of a number read as input. Also draw the corresponding flow chart. [OPTIONAL]
4. Write an algorithm to determine whether a given number is prime or not. Also draw the corresponding flow chart.
5. Write an algorithm to generate first 100 prime numbers. Also draw the corresponding flow chart.
6. Draw a flowchart to input three numbers in the variables a, b and c and hence to find the roots of the quadratic equation $ax^2 + bx + c = 0$. Consider carefully the zero input values of the coefficients a, b and c. [OPTIONAL]
7. Draw a flowchart to input 10 numbers, sort the numbers in ascending order and hence to write the numbers in sorted order. [OPTIONAL]

Console I/O and Conditional Statements

(Any Ten assignments)

1. Write a C program that reads two values from the keyboard, swaps their values and prints out the result.
2. The length and breadth of a rectangle and radius of a circle are input through the keyboard. Write a program to calculate the area and perimeter of the rectangle, and the area and circumference of the circle.
3. If a three-digit integer is input through the keyboard, write a program to calculate the sum of its digits. (Hint: Use the modulo operator '%')
4. An integer is entered as an input through the keyboard. Write a program to find out whether it is an odd number or an even number.
5. Input two integer numbers and divide the larger number by the smaller one. Then display the result using printf() function as a fractional number first and then as a real valued number. (Example: 9 divided by 5 shall yield "4/5" and "1.8" respectively.)
6. Write a C program to find the maximum and minimum of three numbers.
7. Accept three integer numbers and find their average. Next display which numbers are below and above the average value.
8. Temperature of a city in Fahrenheit degrees is input through the keyboard. Write a program to convert this temperature into Centigrade degrees.
9. Write a C program which accepts basic salary as input and prints the gross salary, which is sum of the basic, dearness allowance (60% of basic salary), and house rent allowance (15% of basic salary).
10. A cashier has currency notes of denominations 10, 50 and 100. Write a C program which accepts an amount to be withdrawn, and prints the total number of currency notes of each denomination the cashier will have to give to the withdrawer. [OPTIONAL]
11. If the marks obtained by a student in five different subjects are input through the keyboard, find out the aggregate marks and percentage marks obtained by the student. Assume that the maximum marks that can be obtained by a student in each subject is 100. Input error should be checked. [OPTIONAL]
12. The length and breadth of a rectangle are input through the keyboard. Write a programme to determine (i) radius of a circle and ratio of perimeters of the rectangle and the circle if the areas of the rectangle and the circle are equal (ii) radius of the circle and the ratio of the areas of the rectangle and the circle if the perimeters of the rectangle and the circle are equal. [OPTIONAL]

13. In a town, the percentage of men is 52. The percentage of total literacy is 48. If total percentage of literate men is 35 of the total population, write a program to find the total number of illiterate men and women if the population of the town is 80,000. [OPTIONAL]
14. If a five-digit integer is input through the keyboard, write a program to print a new number by adding one to each of its digits. For example if the number that is input is 12391 then the output should be displayed as 23402. [OPTIONAL]
15. Write a program to find the grace marks for a student using switch. The user should enter the class obtained by the student and the number of subjects he has failed in.
- If the student gets first class and the number of subjects he failed in is greater than 3, then he does not get any grace. If the number of subjects he failed in is less than or equal to 3 then the grace is of 5 marks per subject.
 - If the student gets second class and the number of subjects he failed in is greater than 2, then he does not get any grace. If the number of subjects he failed in is less than or equal to 2 then the grace is of 4 marks per subject.
 - If the student gets third class and the number of subjects he failed in is greater than 1, then he does not get any grace. If the number of subjects he failed in is equal to 1 then the grace is of 5 marks per subject. [OPTIONAL]
16. Read a five-letter word into the computer, then encode the word on a letter-by-letter basis by subtracting 30 from the numerical value that is used to represent each letter. Thus if the ASCII character set is being used, the letter a (which is represented by the value 97) would become a C (represented by the value 67), etc. Write out the encoded version of the word. [OPTIONAL]
17. Read into the computer a five-letter word that has been encoded using the scheme described above. Decode the word by reversing the above procedure, then write out the decoded word. [OPTIONAL]
18. Any year is input through the keyboard. Write a program to determine whether the year is a leap year or not. (Hint: Use the % (modulus) operator)
19. Write a program to check whether a triangle is valid or not, when (i) the three angles of the triangle are entered through the Keyboard (ii) three sides of the triangle are entered through the keyboard. [OPTIONAL]
20. Given three points (x_1, y_1) , (x_2, y_2) and (x_3, y_3) , write a program to check if all the three points fall on one straight line. [OPTIONAL]
21. Given the coordinates (x, y) of a center of a circle and its radius, write a program which will determine whether a point lies inside the circle, on the circle or outside the circle. (Hint: #include <math.h>. Use sqrt() and pow() functions)

22. Any character is entered through the keyboard, write a program to determine whether the character entered is a capital letter, a small case letter, a digit or a special symbol.
23. Given as input an integer number of seconds, write a program to print as output the equivalent time in hours, minutes and seconds. Recommended output format is something like 7322 seconds is equivalent to 2 hours 2 minutes 2 seconds.
24. Write a program which accepts two number X, Y and creates a third number Z by appending Y after X. Example: if X=12 and Y=345 then Z=12345. [OPTIONAL]
25. A certain grade of steel is graded according to the following conditions:

Hardness must be greater than 50

Carbon content must be less than 0.7

Tensile strength must be greater than 5600

The grades are as follows:

Grade is 10 if all three conditions are met

Grade is 9 if conditions (i) and (ii) are met

Grade is 8 if conditions (ii) and (iii) are met

Grade is 7 if conditions (i) and (iii) are met

Grade is 6 if only one condition is met

Grade is 5 if none of the conditions are met

Write a program, which will require the user to give values of hardness, carbon content and tensile strength of the steel under consideration and output the grade of the steel. [OPTIONAL]

Loops

(Any ten assignments)

1. Write a C program which prints all integers divisible by n between 1 and 100 where value of n is provided by the user.
2. Write a C program to find out sum of digits of a given number.
3. Write a C program to find the reverse of a given number. [OPTIONAL]
4. Write a C program to find out sum of the following series.

$$S=1+2+3+4+ \dots +n$$

5. Write a C program to find out sum of the following series.

$$S=1.2+2.3+3.4+4.5+ \dots +n.(n+1)$$

6. Write a C program to find out factorial of a given number.
7. C program to find out sum of the following series.

$$S=1!+2!+3!+4!+ \dots +n!$$

8. Write a C program to find out sum of the following series.

$$S = 1@ + 2@ + 3@ + 4@ + \dots + n@$$

where, $n@$ is the sum of all factors of n . Example: $6@ = 1+2+3+6 = 12$ [OPTIONAL]

9. Write a C program which prints all prime numbers between 1 and 100.
10. Write a program which accepts a number n and prints all prime factors of n . [OPTIONAL]
11. Write a program to generate all combinations of digit 1, 2 and 3 using for loop. [OPTIONAL]
12. Write a program to print the multiplication table of the number entered by the user. The table should get displayed in the following form.

$$29 * 1 = 29$$

$$29 * 2 = 58$$

... ..

13. Write a menu driven program to accept a number in any Number System [from Binary, Decimal, Octal, and Hex] and convert and display the same in any other amongst these.

14. Write a menu-driven program for displaying log-series results for an input number for different log bases. [OPTIONAL]

15. Write a program named SINE to find the sine of an angle. The angle and its unit (degree, radian or grade) should be provided as command line arguments. For the units, short forms as d/D (for degree), r/R (for radian) or g/G (for grade) may be used. The program should use the series

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots,$$

for evaluation. Take care of negative angles and angles in all the quadrants.
[OPTIONAL]

16. Write a C program which accepts 100 integers and displays the counts of positives, negatives and zeros entered.

17. Write a C program to print the first n numbers of the Fibonacci sequence. The Fibonacci sequence is constructed by adding the last two numbers of the sequence so far to get the next number in the sequence. The first and second numbers of the sequence are defined as 0 and 1. We get:

0, 1, 1, 2, 3, 5, 8, 13, 21...

18. Write a program to print out all Armstrong numbers between 1 and 500. If the sum of cubes of each digit of the number is equal to the number itself, then the number is called an Armstrong number. For example, $153 = (1^3) + (5^3) + (3^3)$. [OPTIONAL]

19. Write a C program which prints the first 10 happy numbers. If you iterate the process (assume maximum 100 iterations) of summing the squares of the decimal digits of a number and if the process terminates in 1, then the original number is called a Happy number. For example 7 is a happy number as $7 \rightarrow 49 \rightarrow 97 \rightarrow 130 \rightarrow 10 \rightarrow 1$. [OPTIONAL]

20. An important property of square numbers: If a natural number is a square number, then it has to be the sum of Successive Odd Numbers starting from 1.

For example:

Perfect Square	Sum of Odd Numbers
4	1 + 3
9	1 + 3 + 5
16	1 + 3 + 5 + 7
25	1 + 3 + 5 + 7 + 9
36	1 + 3 + 5 + 7 + 9 + 11
49	1 + 3 + 5 + 7 + 9 + 11 + 13
64	1 + 3 + 5 + 7 + 9 + 11 + 13 + 15

Now using this property, find the square root of any perfect square. [OPTIONAL]


```

      +
    +++
  +++++
+++++++
+++++++
+++++
+++
+

```

```

      1
     121
    12321
   1234321
  12321
   121
    1

```

```

+++++++
+++++
+++
++
+
+
++
+++
++++
+++++
+++++

```

```

      +
    + +
  +   +
+     +
+   +
+ +
+

```

[OPTIONAL]

25. Write a program to add first seven terms of the following series using a for loop:

$1/1! + 2/2! + 3/3! + \dots$ [OPTIONAL]