CSE110 Practice Assignment 1

Task 1.1: Draw the flowchart of a program that reads a student's mark for a single subject, and prints out "Pass" if the student got more than 50, and "You shall not pass" otherwise.

Task 1.2: Draw the flowchart of a program that reads a student's mark for a single subject, and prints out the corresponding grade for that mark. The mark ranges and corresponding grades are shown in the table below. Valid range of marks is 0 to 100. Print "Wrong Input" otherwise.

Marks	Grade
90 and above	A
80-89	B
70-79	C
60-69	D
50-59	E
Below 50	F

Task 1.3: Draw the flowchart for printing area of a rectangle given height and width as inputs from the user.

Task 1.4: Draw the flowchart of a program that calculates the tax as follows:

a) No tax if you get paid less than 10,000
b) 5% tax if you get paid between 10K and 20K
c) 10% tax if you get paid more than 20K
d) NO TAX IF YOU ARE LESS THAN 18 YEARS OLD.

Hint: Take payment and age from user as inputs; then calculate tax and prints it.

Task 1.5: Draw the flowchart of a program that finds the number of hours, minutes, and seconds in a given number of seconds. For example, how many hours, minutes, and seconds are there in 10,000 seconds?

Task 1.6: Suppose the following expressions are used to calculate the values of L for different values of S:

L = $3000 - 125S^2$ if S < 100

L = $\frac{12000}{4+(\frac{S^2}{149000})}$ if S >= 100

Draw the flowchart of a program that reads a value of S and then calculates the value of L.

Task 1.7: Draw the flowchart of a program that reads the values for the three sides: x, *y*, and z of a triangle, and then calculates its area. The area is calculated as follows:

area = $\sqrt{s(s-x)(s-y)(s-z)}$ where s is $\frac{x+y+z}{2}$

Task 1.8: A leading newspaper pays all of their freelance writers at a rate of Tk. 500 per published article. Draw the flowchart of a program that will read the number of published articles for one writer, and print the total fees for that writer.

Task 1.9: Draw flowchart of a program to find the largest among three different numbers entered by user.

Task 1.10: Draw the flowchart for the following:

1) Ask the user to enter the name of his favorite car.
2) Display the name of the user's favorite car 4 times.

Example: If the user enters “Toyota”, your program should print the name Toyota 4 times.

Task 1.11: Assume there are two variables a and b. Take Values of these variables from the user.
For example, user gave following two values.
a = 721
b = 463
Then exchange / swap values in such a way so that printing the variable a gives 463 and b gives 721.
Sample Input:
721
463
Sample output:
Before Exchange:
a = 721
b = 463
After Exchange:
a = 463
b = 721

Task 1.12: Take value of a, b, c from the user. Then print in such a way that
value of a goes to b
value of b goes to c
value of c goes to a

Task 1.13: Take value of a, b, c, d from the user. Then print in such a way that
value of d goes to c
value of c goes to b
value of b goes to a
value of a goes to d

Task 1.14: Take a number from user and print its absolute value. If user gives minus five, print positive five. If user gives positive five, print positive five. No need to print plus sign.

Task 1.15: Take hour from the user as input and tell it is time for which meal.
User will input the number in 24-hour format say 14 to mean 2pm, 3 to mean 3am, 18 to mean 6pm etc.
Valid inputs are 0 to 23. Inputs less than 0 or more than 23 are invalid in 24-hour clock.
Input will be whole numbers. For example, 3.5 will NOT be given as input.

Inputs: Message to be printed
4 to 6: Breakfast
12 to 13: Lunch
16 to 17: Snacks
19 to 20: Dinner
For all other valid inputs, say "Patience is a virtue"
For all other invalid inputs, say "Wrong time"

For example,
If the user enters 4, your program should print the message "Breakfast".
If the user enters 5, your program should print the message "Breakfast".
If the user enters 6, your program should print the message "Breakfast".
If the user enters 0, your program should print the message "Patience is a virtue".
If the user enters 1, your program should print the message "Patience is a virtue".
If the user enters 18, your program should print the message "Patience is a virtue".
If the user enters 23, your program should print the message "Patience is a virtue".
If the user enters 24, your program should print the message "Wrong Time".
If the user enters -1, your program should print the message "Wrong Time".
If the user enters 27, your program should print the message "Wrong time".

CSE110 Practice Assignment 2

Task 2.1: Draw the flowchart for the following:

a) Ask the user to enter the name of his favorite car.

b) Ask the user to enter a Number

c) Display the name of the user's favorite car, number of time specified in the second step. Example:

If the user enters "Toyota" and 20, your program should print the name Toyota twenty times.

Task 2.2: Draw the flowchart to display all the odd numbers between 10 and 50.

Task 2.3: Draw the flowchart for the following:

Take seven numbers input from the user and print the maximum and the average.

Task 2.4: Draw the flowchart for the following:

Take eight numbers input from the user and find the minimum and the average of the even numbers entered by the user. [If the user enters odd numbers ignore them]

Task 2.5: Draw the flowchart of a program that reads the value of *n* and calculates the value of y (NOT y^3) if the expression of y is as follows:

$$Y^3 = 1^3 + 2^3 + 3^3 + 4^3 + 5^3 + 6^3 + ...+ N^3$$

Task 2.6: Draw the flowchart of a program that will calculate the value of y if the expression of y is as follows (n is the input):

$$Y = 1^2 - 2^2 + 3^2 - 4^2 + 5^2 ..... + N^2$$

Task 2.7: Draw the flowchart of a program which adds all numbers that are multiples of both 7 and 9 up to 600.

Task 2.8: Draw the flowchart of a program which adds all numbers that are multiples of either 7 or 9 up to 600.

Ensure that numbers like 63 are added only once in the sum.

Task 2.9: Draw the flowchart of a program which adds all numbers that are multiples of either 7 or 9 but not both, up to 600.

Task 2.10: Draw the flowchart of a program that asks the user to enter ten numbers then display ONLY the total and the average of the odd numbers among those ten numbers.

[Hint: Example Input: 1 2 3 4 5 6 7 8 9 10 and Example Output: Total is 25 and Average is 5 (i.e., Total is 25 = (1+3+5+7+9) and Average is 25/5 = 5)]

Task 2.11: Solve Task 10 for even numbers instead of odd numbers.

Task 2.12: Solve Task 10 for numbers that are multiples of 4, instead of odd numbers.

Task 2.13: Write a flowchart of a program that reads a number N, and prints out the sum of all odd numbers from 1 to N inclusive. For instance, if the input is 6, the output for the program should be 9.

Task 2.14: Write a flowchart that reads a list of numbers, and prints out the product of all the numbers read. You may assume that the user first inputs the total number of numbers. For example, if the first input is 4, then the program has to read in four numbers from the user, and print out the product of these four numbers. Assume that user will never enter first number as zero.

Task 2.15:

Write a flowchart that will read 20 numbers from the user, and then print the first number, the sum of the first 2 numbers, first 3 numbers, and so on up to the sum of 20 numbers.

CSE110 Practice Assignment 3

Task 3.1: Draw the flowchart of a program that reads marks of ten courses and prints the maximum, minimum and average of those ten marks.

Task 3.2: In mathematics, the Fibonacci numbers form a sequence defined by the following recurrence relation:

$$F(n) := \begin{cases} 0 & \text{if } n = 0; \\ 1 & \text{if } n = 1; \\ F(n-1) + F(n-2) & \text{if } n > 1. \end{cases}$$

That is, after two starting values, each number is the sum of the two preceding numbers. The first Fibonacci numbers, also denoted as Fn, for n = 0, 1, … , are:
0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233,
Draw the flowchart of a program, which prints all Fibonacci numbers that are less than 1600.

Task 3.3: Draw flowchart of a Java program which takes a number and tells how many digits are in that number. Example: if user gives 9876, you should print 4.

Hint: keep dividing by ten and count how many times this could
be divided. 9876 by 10, is 987, count that got 1 digit

987 by 10, is 98, count that got 1 digit (total 2)
98 by 10, is 9, count that got 1 digit (total 3)
9 by 10, is 0, count that got 1 digit (total 4)
done!

Task 3.4: Draw flowchart of a Java program which takes a number and prints the value of 10 to the power that number. You need to use loop because variable in the power isn't allowed.
For example: if user gives 3, print 1000.
Hint: Keep multiplying 1 by 10, again and again, 3 (or
n) times like sum=sum+n, you need to write,
product = product x 10
1 x 10 = 10
10 x 10 = 100
100 x 10 = 1000

Task 3.5: Draw flowchart of a Java program which takes a number and prints the digits from unit place, then tenth, then hundredth, etc.
Example: if user gives 32768, then print 8, 6, 7, 2, 3
Hint: Taking remainder/modulus of division by 10.
After printing the remainder, drop the last digit by dividing by 10. Then start over.
32,768 % 10 = 8
32,768 / 10 = 3, 276

3, 276 % 10 = 6
3, 276 / 10 = 327

327 % 10 = 7
327 / 10 = 32

32 % 10 = 2
32 / 10 = 3
3 % 10 = 3
3 / 10 = 0

Task 3.6: Draw flowchart of a Java program which takes a number and prints the digits from left to right. Example: if user gives 32768, then print 3, 2, 7, 6, 8
Hint: count how many digits
calculate 10 to the power that (number of digits) minus 1.

Say, 32768 has 5 digits, so you calculate 10 to the power 4
which is 10,000. Then divide 32,768 by 10,000 and thus you get
3.

take remainder of 32,768 by 10,000 and thus you
get 2,768 Then divide 10,000 by 10 to get 1,000

Then divide 2,768 by 1,000 and thus you get 2.

take remainder of 2,768 by 1,000 and thus you get 768
keep going on until there is no more digits left (zero!).

In short:
Loop 1: First count digits, say 5 in this case for 32,768
Loop 2: Then calculate 10 to the power 4 (5-1), that is 10,000.
Loop 3: Then repeat following three steps

32,768 / 10,000 = 3
32,768 % 10,000 = 2,768
10,000/10 = 1,000

2,768 / 1,000 = 2
2,768 % 1,000 = 768
1,000/10 = 100

768 / 100 = 7
768 % 100 = 68
100/10 = 10

68 / 10 = 6
68 % 10 = 8
10/10 = 1

8 / 1 = 8
8 % 1 = 0
1/10 = 0

Task 3.7: Telling words: Draw flowchart of a program that takes a number between 0 and 9. You have to print that number in words.
Hint: if n == 1, then print "one"
else if n == 2, then print "two"

Task 3.8: Combine Task 6 and 7 into a single flowchart so that it can tell any number in words. Example: If user gives 932, print nine three two.

Task 3.9: Draw flowchart of a program that takes a number and prints all numbers up to that number. If the user gives 8, print 1 to 8.

Task 3.10:
Draw flowchart of a program that takes a number and counts how many times that number can be divided by all numbers up to that number (Those numbers are also known as factors)
If the user gives 8, tries to divide 8 by each of 1 to 8 and count how many times it could be divided.
For example:
If user enters 8,
try to divide 8 by 1, its divisible (increase count to 1)
try to divide 8 by 2, its divisible (increase count to 2)
try to divide 8 by 3, its NOT divisible
try to divide 8 by 4, its divisible (increase count to 3)
try to divide 8 by 5, its NOT divisible
try to divide 8 by 6, its NOT divisible
try to divide 8 by 7, its NOT divisible
try to divide 8 by 8, its divisible (increase count to 4)
Now print the count which is 4 in this case.

Task 3.11: If a number is NOT divisible any number other than 1 and itself, then it is called prime number.
For example, 13 is a prime number because it is NOT divisible by any number other than 1 and 13 (itself). Take one number from the user and tell if it is prime number or not.

Hint: Use the technique from Task 25 and count factors of the input. Factors are those numbers between 1 and n that can divide the number, n. If there are more than two factors (1 and n), then the number, n is not prime because it was divisible by other numbers.

Task 3.12: Write flowchart of a program that finds and displays all the prime numbers less than 1000.
Task 3.13: Modify Task 11, instead of counting factors, print sum of factors.

Task 3.14: Modify Task 11, calculate sum of factors less than the number itself. If the sum equals to the number, then print that the number is a perfect number.

Example 1: User enters n = 6. Factors of 6 are 1, 2, 3. Sum of those factors 1+2+3=6 which is same as the number 6(n). So, print that 6 is a perfect number.

Example 2: If user enters 8. Factors of 8 are 1, 2, 4. Sum of those factors 1+2+4=7 (NOT equal to 8). So, print that 8 is NOT a perfect number.

Task 3.15: Ask user for a range. Count how many numbers are prime number and how many numbers are perfect numbers between that range.
For example, between 2 and 6 there are 3 prime numbers (2, 3, 5) and 1 perfect number (6).

Sample Input: 2 6
Sample Output:

Between 2 and 6,
Found 3 prime numbers
Found 1 perfect number.