CS 11 Exercise 1
1st Semester, AY2018-2019

August 13, 2018

1 Instructions

- Work on this assignment with your group. Group assignments are available in UVLe.
- Your solution must be in PDF format. Submit your solutions via UVLe. Only one of the group members must submit the solution.
- The solution for each item must start on a new page.
- Make sure that your algorithm works given the sample input and output. You must also check if your algorithm can also handle input other than the ones given.
- Submit your solutions on or before Saturday, August 18 at 11:59pm.

2 Problems

1. **Fibonacci Numbers (4 points)** - Given an integer $n \geq 0$, draw a flowchart for printing the Fibonacci numbers from F_0 to F_n. For $i \geq 2$, the ith Fibonacci number F_i is $F_{i-1} + F_{i-2}$, with $F_0 = 0$, and $F_1 = 1$.

2. **Approximating e (4 points)** - Given a real number x, the value of e^x is equal to the following summation:

$$\sum_{n=0}^{\infty} \frac{x^n}{n!} = \frac{x^0}{0!} + \frac{x^1}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \frac{x^5}{5!} + \cdots$$

   Draw a flowchart for approximating e^x by adding the terms of the summation. Add all the terms $\frac{x^i}{i!}$ such that $\left|\frac{x^i}{i!}\right| > 0.000001$

3. **Largest Element (4 points)** - Given $n \geq 0$ and a list of n numbers $A_0, A_1, A_2 \ldots, A_{n-1}, A_n$, write an algorithm for getting the largest element in the list.

4. **Prime Numbers (4 points)** - Given $n \geq 0$, draw a flowchart for printing the prime numbers in the interval $[0, n]$.

   (Hint: you can use the *modulo* operator (denoted by %) to get the remainder when two numbers are divided. For example, $5\%2 = 1$ since dividing 5 by 2 will give you a remainder of 1. Another example is $13\%5 = 3$, since you will get a remainder of 3 when you divide 13 by 5.)

5. **Sum of Digits (4 points)** - Given a positive integer N with an arbitrary number of digits, reduce N by repeatedly computing the sum of its digits until arriving at a single-digit number. For example, if our input N is 1234567, its reduction will be equal to 1 since

   (a) $1 + 2 + 3 + 4 + 5 + 6 + 7 = 28$

   (b) $2 + 8 = 10$

   (c) $1 + 0 = 1$

3 Sample Input and Output

1. **Fibonacci Numbers**

Input	Output
0	0
1	0 1
2	0 1 1
10	0 1 1 2 3 5 8 13 21 34 55
-1	(no output must be printed)

2. **Approximating** e

Input	Output
0	1.0
1	2.718281 . . .
2	7.38905 . . .
3	20.085536 . . .
-1	0.367879 . . .

3. **Largest Element**

Input	Output
1; -100	-100
5; 1 2 3 4 5	5
10; 27 23 44 44 21 91 69 97 43 3	97
0;	(no output must be printed)
-1;	(no output must be printed)

4. **Prime Numbers**

Input	Output
0	(no output must be printed)
1	(no output must be printed)
2	2
10	2 3 5 7
30	2 3 5 7 11 13 17 19 23 29
-1	(no output must be printed)

5. **Sum of Digits**

Input	Output
0	0
123	6
2018	2
1234567	1
43674566234	5
-1	(no output must be printed)
-199	(no output must be printed)