

High School Coding Contest Saint Anselm College
Saturday, April 2, 2016 @9:00-11:00 AM
Contest Problems
I code therefore I am!

General: we do not test for invalid input

Problem 1. Check if anagram

1. Write a program that takes an input consisting of two strings (input includes only lower case letters without blanks) and checks if one input is the permutation of the other (anagram).

Example1.	Input: Enter two words? people lepoee	Output: false
Example2.	Input: Enter two words? apple papel	Output: true
Example3.	Input: Enter two words? dan dna	Output: true

Problem 2. Parentheses alignment

2. Write a program that only accepts strings that consists of matching parentheses: () , { } or []. A pair of matching parentheses can be nested/contained within another matching pair. For example `[[]]`, `{{()}}` are acceptable but `[{]}` is not.

Example1.	INPUT: Enter string? ()	OUTPUT: false
Example2.	INPUT: Enter expression? ({}[{}])()	OUTPUT: true
Example3.	INPUT: Enter string? ()))	OUTPUT: false
Example4.	INPUT: Enter expression? ({}[{}])	OUTPUT: true
Example5.	INPUT: Enter string? (([])	OUTPUT: false

Problem 3. Count ZEROS

3. Write a program to calculate the number of zeros at the end of the resulting number when you calculate $n!$, where $n! = 1 * 2 * 3 * .. (n-1) * n$. The range of n is $1 - 10^9$.

Example1.	INPUT: Enter number? 12	OUTPUT: 2
-----------	-------------------------	-----------

Explanation: $12! = 479,001,600$ (i.e. ends is 2 zeros)

Example2.	INPUT: Enter number? 200	OUTPUT: 49
Example3.	INPUT: Enter number? 12345	OUTPUT: 3082

Problem 4. Check collision

4. You will create two rectangles in 2D. Your input will consist of the coordinates of the upper left corner (x,y), width, and height of the rectangles. Write a program that checks if the rectangles touch/overlap. If they touch/overlap the program will return "collision", otherwise it will return "No collision".

Example. INPUT: Enter rectangle1: -2 1 4 3
Enter rectangle2: -5 -1 5 2
OUTPUT: **collision**

INPUT: Enter rectangle1: 1 3 2 5
Enter rectangle2: -1 -1 4 1
OUTPUT: **collision**

INPUT: Enter rectangle1: -1 1 1 1
Enter rectangle2: 0 1 1 1
OUTPUT: **collision**

INPUT: Enter rectangle1: 0 1 1 1
Enter rectangle2: 3 3 1 1
OUTPUT: **No collision**

Problem 5. How many ways

5. Write a program that returns the number of ways you can write a positive integer as a multiplication of its factors.

6.

Example1. INPUT: Enter number? 24 OUTPUT: Total=7

Explanation: 1x24, 2x12, 3x8, 4x6, 2x2x6, 2x3x4, 2x2x2x3

Example2. INPUT: Enter number? 100 OUTPUT: Total=9

Explanation: 2x2x5x5, 2x2x25, 2x5x10, 2x50, 4x5x5, 4x25, 5x20, 10x10, 1x100

Example3. INPUT: Enter number? 2016 OUTPUT: Total=171