

Lab 02 - Mathematical Concepts

Instructions:

- When studying mathematics, one is introduced to several mathematical structures that contain several properties, and are used as the building blocks for mathematical concepts and theorems.
- Your objective is to define functions that identify if a data structure represents a particular mathematical structure or retrieve properties of a mathematical structure.
- Your source codes must compile and can only include the libraries 'iostream', 'iomanip', 'string', and user-defined libraries from the lab to receive any credit.
- A cumulative task will not receive credit if the required previous tasks are not completed.
- Your submissions must be submitted to the GitHub repository in the Lab02 directory.
- Cheating of any kind is prohibited and will not be tolerated.
- Violating or failing to follow any of the rules above will result in an automatic zero (0) for the lab.

Grading

| Task | Maximum Points | Points Earned |
|-------|----------------|---------------|
| 1 | 1 | |
| 2 | 2 | |
| 3 | 1 | |
| 4 | 1 | |
| Total | 5 | |

Note: solutions will be provided for tasks colored blue only.

Task 1

- Create a C++ file named `'task01.cpp'` that defines and tests a Boolean function named `isSet()` that takes an array parameter and an integer parameter. Given that the integer parameter represents the size of the array parameter, it returns true only if the array represents a set; otherwise, it returns false.

Example:

The invocations `isSet({1,2,3,4},4)` and `isSet({4,2,3,4},4)` will return true and false, respectively.

Task 2

- Create a C++ file named `'task02.cpp'` that defines and tests a Boolean function named `isPermutation()` that takes two array parameters and an integer parameter. Given that the integer parameter represents the size of both array parameters, it returns true only if the arrays form a permutation; otherwise, it returns false.

Example:

The invocations `isPermutation({1,2,3,4},{2,1,3,4},4)` and `isPermutation({4,2,3,4},{2,3,4,4},4)` will return true and false, respectively.

Task 3

- Create a C++ file named `'task03.cpp'` that defines and tests an integer function named `degree()` that takes a two-dimensional Boolean array parameter and two integer parameters. Given that the first integer parameter represents the size of both dimensions of the array parameter (an adjacency matrix), it returns the sum of out-degree and in-degree of the vertex (the second integer parameter). An edge (from, to) implies (row, column). A self-loop is both an out-degree and an in-degree.

Example:

The invocation `outDegree({{1,0,1},{0,1,1},{1,0,0}},3,1)` returns 3.

Task 4

- Create a C++ file named `'task04.cpp'` that defines and tests the Boolean functions `sum()` and `carry()` that both take three Boolean parameters. The function `sum()` returns the binary result of the sum of the parameters. Meanwhile, the function `carry()` returns the binary carry of the sum of the parameters.

Example:

The invocations `sum(true,false,false)` and `carry(true,false,false)` will return true and false, respectively.

Extra Credit

- Prove using induction that

$$\sum_{i=1}^n \frac{1}{4i^2 - 1} = \frac{n}{2n + 1}$$

(1 point)