

Department of Computer Science Graduate Exam in Programming Spring 2019

You must write **Java** or **C++** on the first page of your answers!! You may answer the questions using either Java or C++, but you must use the same language for **each** question on the exam.

1. (2 pts) Write a function named `bar` that takes an array of integers and a size as an argument. Each value in the array represents an amount of sales for a month. For each value in the array, the function should output a line in a bar graph. Each asterisk in the output represents a full \$100 in sales (fractional amounts are not represented).

For example, for the array {120, 250, 595} with size 3 it should output:

```
*
**
*****
```

The next question uses the following class, which implements a binary search tree:

<pre>// C++ class BST { private: struct TreeNode { int value; TreeNode *left; TreeNode *right; }; //points to the top node TreeNode *root; public: BST() {root=NULL;} bool find(int x); }</pre>	<pre>// Java class BST { private class TreeNode { int value; TreeNode left; TreeNode right; } //refers to the top node private TreeNode root; public BST() {root = null;} public boolean find(int x){...} }</pre>
--	--

2. (2 pts) Write the member function `find(int x)` which returns true if the value `x` is in the tree, otherwise false. A binary search tree has the following property: at *each* node in the tree, the elements in the left subtree are less than the node value, and the elements in the right subtree are greater than the node value. Your solution should take advantage of this property to avoid performing an exhaustive search of the tree. You may add new private functions to the class as needed.

3. (2 pts) Write a **recursive** function named `replace` that takes an array of integers, its size, and two integers `x` and `y`, and replaces all the occurrences of `x` in the array with `y`. For example, `replace(arr, 5, 0, 1)` where `arr` is `{2, 0, 5, 6, 0}` should change `arr` to `{2, 1, 5, 6, 1}`.

Do not use loops, extra parameters, or global or static variables.

4. (4 pts) A line in two dimensional space may be defined using two points (x_1, y_1) and (x_2, y_2) where x_1, y_1, x_2 and y_2 are real numbers. If $y_1 = y_2$, then it is a horizontal line, and if $x_1 = x_2$, then it is a vertical line. If the line is not vertical, then the slope of the line is $(y_2 - y_1) / (x_2 - x_1)$. The slope of a vertical line is ∞ (undefined). Two lines are parallel if they have the same slope or if both are vertical lines.

Design and implement the `LineType` class to represent a line as follows:

- A. Implement a class (or struct) called `Point` to represent a point (x, y) . The `x` and `y` members may be public. You may nest the declaration of `Point` inside of `LineType`, but it is not required.
- B. Implement a class called `LineType` to represent a line. The only member variables should be two variables of type `Point`. Include the following functions in your `LineType` class:
- a 4-argument constructor that takes four values to specify the 2 points (you may assume the provided values will yield two different points).
 - `isHorizontal` to return true if the line is horizontal, false otherwise.
 - `isVertical` to return true if the line is vertical, false otherwise.
 - `slope` to compute and return the slope (for non-vertical lines).
 - `parallel` to return true if the line is parallel to another line, false otherwise.

It is ok for `slope` to cause a runtime error if the line is vertical.

It is NOT ok for `parallel` to cause a runtime error.