

CompSci 1, Fall 2005, Midterm Bonus

Name _____ Login _____ Lab Section _____

Signature acknowledging Duke Community Standard _____

Resources used:

Hours worked:

Points earned on these problems will be counted toward your Midterm score. It's not possible to earn more than 100% on a test.

You may use your books and notes in solving these problems. You may not use the Internet, you may not use Eclipse, and you may not talk to anyone. For questions, please use the discussion forum for clarifications/issues.

You must turn in your work on paper in class. Indicate how many hours you worked on the problem and what resources (book, notes) you consulted in doing the problems. Your submission of the work with your signature indicates your adherence to the Duke Community Standard.

You may also turn in hard-copy rather than submitting your work. No submissions, hard-copy or electronic, will be counted if received after 2:20pm on Friday, October 14.

1. The following subroutine returns the sum of the numbers between two integers (inclusive).

```
public int SumBetween(int begin, int end)
{
    int sum = 0;
    int k = begin;

    while (k <= end)
    {
        sum = sum + k;
        k = k + 1;
    }
    return sum;
}
```

Write the necessary code to declare an integer variable `c` and assign to it the sum of the numbers between 10 and 20 by calling `SumBetween`. (2 points)

2. Write a subroutine called `sumUp` that returns the sum up to an integer. As an example, the following code calls `SumUp`.

```
int c;  
c = SumUp(5);
```

The value in `c` would be 15 ($1 + 2 + 3 + 4 + 5$). Write `SumUp` below. You can, of course, use `SumBetween`. (3 points)

3. Write a method `numDistinct` that takes three integers as parameters and that returns the number of distinct integers among the three.

For example, if the following call is made: `numDistinct(18, 3, 4)` the method should return 3 because the parameters represent 3 different numbers (the values 18, 3 and 4).

By contrast, the following call: `numDistinct(6, 6, 6)` would return 1 because there is only 1 distinct number among the three parameters (the value 6). The values passed to your method might appear in any order whatsoever, so you cannot assume that duplicate values would appear together.

For example, if the following call is made: `numDistinct(7, 31, 7)` the method should return 2 because there are 2 distinct numbers among the parameters (the values 7 and 31).

Write your solution to `numDistinct` below. (5 points)