CSC 211: Introductory Programming and Design

Midterm Exam 1 – October 08, 2003

Duration of the examination: 1 h 15 min.

Documentation: Textbook and notes allowed.

Comments: Use your time wisely (no problem deserves that you spend 1/2h on it).
Problem 3 can be stressful. Avoid keeping it for the last 5 minutes of the exam.
Don’t overcheck your answer for that problem either: Chances are you will only add new errors.

Problem 1 15 Pts

Identify as many error as you can in the following code, and indicate which are syntax errors (as opposed to errors of logic).

```java
public void myMethod(int m, float val);  
{
    int m, n, k = 6;
    float x = 3.5;

    n += 5;
k += k++ + ++k;
if (k > 10)
    m = k / 3;
    n *= 2;
else if (0 < k < 5);
    m = -m
if (k = 0)
    m = 5;
x + val = m ** 2 + n;
return x;
}
```
Problem 2 20 Pts

Calculate the value taken by variable \( x \) resulting from the execution of the code fragment below. Write parentheses to indicate the order of evaluation of the expression, according to the precedence and associativity rules of Java, and indicate all the type conversions, implicit or explicit, that occur during the evaluation. The more information you provide on your computations, the more you increase your chances of getting partial credit if your final result is incorrect.

```java
int m = 6, n = 21;
double x = 10.0;
float z = 1.0F;

x -= (float) n-- / m * -3 + n / - m / 3.0 + z * ++n / 3 * - ++x;
```
Problem 3

Given the following (admittedly ugly and messy) method definition

```java
public static Integer myMethod(int a, Integer theInt)
{
    a++;
    theInt = new Integer(theInt.intValue() + ++a);
    return new Integer(a + theInt.intValue());
}
```

Trace the value of \(a\), \(b\), \(m\), \(n\), and \(p\) during the execution of the following block of code. Feel free to use the blank sheet on the next page. In general, the more information you provide on your thought process, the more you increase your chances of getting partial credit if your final result is incorrect.

```java
int a=2, b=5;
Integer m = new Integer(6),
    n = new Integer(10),
    p;

n = myMethod(a, m);
p = myMethod(++b, m);
p = myMethod(a + ++b, new Integer(3));
m = myMethod(n.intValue() + 1, m);
```
Problem 4 25 Pts

Write on one of the attached blank pages the code of a method named stringSubstitution that takes as parameters three String objects: the input string, the target string, and the replacement string. Your method should return a new String object whose value is that of the input string where all occurrences of the target string have been replaced by the replacement string.

If the input string is “Just say: A cup of Java a day keeps the C++ away”, the target string is “ay”, and the replacement string is “–@*@–”, then your method should return the following string:

“Just s–@*@–: A cup of Java a d–@*@– keeps the C++ aw–@*@–”.

Note: I should not have to remind you that Appendix E of our textbook gives a good summary of the API of the String class. Here again, I am not asking you to come up in such a limited time with a clever, efficient algorithm, but only with one that will compute the correct value. On the other hand, I definitely expect you to write correct, readable, properly indented Java code.
Problem 5  
Properly indent on the grid provided in the next page the code listed below. You can select any of the three standard indentation styles we saw in class, but you have to be consistent in your use of that style.

```c
while (x > N)
    y = x + 1;
    if (y > 10) {
        x = x + 2;
        n = y + 1;
    }
else
    y = x - 1;
    n = 2 * y;
if (x*x < 50)
    { n = 15;
    y = x + 2;
    if (y > n)
        if (y <100)
            if (x + y > 0)
                while (x < 100)
                x = x + 3;
        else
            if (n > Z)
                x = 20;
            y = n + 1;
    }
    x = x + 2;
```