CS 115 Midterm 1 Review Quiz

February 26, 2009

To receive full credit for the review quiz:

- Show me your work on the quiz, either during or immediately after today's class. You work in groups of three and show me just one quiz for the group. If you work in a larger group, you will need to show me one quiz for every three students.
- Be willing and able to participate in the discussion of the quiz.

The rules below will apply to the actual midterm. To make the most of the available test time, you should read and understand them in advance.

Rules

- You may use one handwritten 8.5 x 11" cheat sheet (front and back). This is the only resource you may consult during this exam.

- You must show your work/explain your answers in order to receive partial credit for incorrect answers.

- All snippets of code can be assumed to be enclosed within `int main()`. You can assume that the `iostream`, `fstream`, `iomanip`, `string`, and `cmath` libraries have been included at the beginning of the program.

- When you are asked to write *a snippet* of code, you may also assume that it is enclosed within `int main()` and that any necessary libraries have been included.

- When you are asked to write *a complete program*, you must write the `#include` statements, the `int main()`, etc. in your solution to receive full credit.

- A line consisting solely of "..." represents one or more unspecified C++ statements, some of which may change the values of program variables.

Problem 1: 25 points.

What is the output of each of the following segments of code?

(a)

```
int a = 2;
cout << a;
```

(b)

```
int a = 2;
cout << "a";
```

(c)

```
int a = 2.5;
cout << a;
```

(d)

```
bool b = (5 > 4) && (2 != 0);
cout << b;
```

(e)

```
int a = 99;
cout << a++;
```

(f)

```
if (5 % 4) {
     cout << "Baa! ";
}
else {
     cout << "Moo! ";
}
```

(g)

```
int x = 1;
do {
     cout << "Woof!";
} while (x > 1);
```

(h)

```
for (int i = 0; i < 2; i++) {
     cout << i*2 << ' ';
}
```

Problem 2: 25 points.

State whether each segment of code is valid C++. If it is not valid C++, fix it so that it will compile. (Valid C++ means that it will compile without errors.)

(a)

```
int i;
…
if (i = 0) {
     cout << "Neigh!";
}
```

(b)

```
int i;
…
if (i <0 && > -2) {
     cout << "Oink! ";
}
```

(c)

```
for (int i=5; i >= 0; i--) {
     cout << "Meow! "
}
```

(d)

```
int i;
…
if (i > 0) {
     cout << "Positive!";
}
else if (i < 0) {
     cout << "Negative!";
}
else (i ==0) {
     cout << "Zero!";
}
```

(e)

```
while (true) {
     cout << "Arf!" << endl;
}
```

Problem 3: 25 points.

Write short snippets of code to accomplish the following tasks:

(a) Input an integer value from the user. If the user's integer was valid, print it out to the screen.

(b) Given three integer variables a, b, and c (assume these are already declared and defined), print

`Unique!`

if they are 3 different integers and

`Not unique!`

otherwise.

(c) Given the same 3 integer variables a, b, and c, compute and print their average (as a floating-point number).

(d) Given an integer variable N (assume N is already declared and defined), print
`Odd`
if N is odd and
`Even`
if N is even.

(e) Repeatedly ask the user to type in a word, and print that word back to the screen. If the user types
`quit`
do not print that word back to the screen, and do not ask the user to type in any more words.

Problem 4: 25 points.

For this problem, you must write a **complete program** that does the following:

- Computes the sum of the integers between 1 and 1000 inclusive and prints the sum out.
  (Note: please use a loop to add the numbers rather than using a mathematical shortcut.)

- Computes the product of the numbers between 1 and 2 inclusive, counting in steps of 0.05, and prints the product on a new line.