CS 115 Exam 1, Spring 2010

Your name: _____

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.
- Explain/show work if you want to receive partial credit for wrong answers.
- All snippets of code can be assumed to be enclosed within `int main()`. You can assume that the `iostream` and `cmath` libraries have been included at the beginning of the program.
- To write *a snippet* of code, you may also assume that your code is enclosed within `int main()` and that any necessary libraries have been included. In a snippet of code, *DO NOT* use return statements to end the program!
- 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 C++ statements, some of which may change the values of program variables.

Grade (instructor use only)

	Your Score	**Max Score**
Problem 1		24
Problem 2		16
Problem 3		30
Problem 4		30
Total		100

Problem 1: 24 points.

What will print to the screen when each of the following snippets of code is executed?

(a)

```
cout << (pow(2, 3))/2 << endl;
```

(b)

```
int x = 4;
cout << x << endl;
```

(c)

```
int a = 17;
int b = 5;
cout << a % b << endl;
```

(d)

```
int x;
cin >> x;
while (x > 0) {
     x--;
}
while (x < 0) {
     x++;
}
cout << x << endl;
```

(e)

```
int a = 2;
if (a > 0) {
     a = -1 * a;
}
else if (a < 0) {
     a = 3;
}
cout << a << endl;
```

(f)

```
for (int i=3; i > 0; i--) {
     cout << i << " ";
}
```

Problem 2: 16 points.

State whether each snippet of code is valid C++, meaning that it will compile. If it is valid, write "valid." If it is invalid, write "invalid" and make a small change so that it will compile.

(a)

```
int a;
cin >> a >> endl;
cout << a << endl;
```

(b)

```
int i = 5;
do while {
     cout << i << endl;
} (i > 7);
```

(c)

```
int q = 2;
cout << (2*q)++ << endl;
```

(d)

```
for (int i=1; i < 0; i++) {
     cout << "Howdy!\n";
}
```

Problem 3: 30 points.

Write short snippets of code to accomplish the following tasks:

(a) *Assume:*

- An integer variable `N` has already been declared and defined.

Your task:

- Print the square root of every number between 1 and `N`, inclusive. If `N` is less than 1, do not print anything.

(b) *Assume:*

- Float variables `A`, `B`, and `C` have already been declared and defined.

Your task: Print the *mode* of this data. That is:

- If two or more of the variables' values are equal to each other, print their value exactly once.
- If all 3 values are different, print
  `No mode.`

(c) *Your task*:

- Prompt the user to type their name.
- If they type "none", do not greet them by name and do not print anything else.
- If they type anything else, greet them by name and repeat these steps.
- Do not use a `return` statement in this snippet of code!

(d) *Assume:*

- The float variable `s` has already been declared and defined.

Your task:

- Compute and print the area of a square whose side length is `s` (remember that area is the square of the side length).
- Format your answer EXACTLY this way:

```
The area of a square with side length _____ is ______.
```

Problem 4: 30 points.

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

- Prompts the user to enter 1000 integers.
- If the user enters something that cannot be read as an integer, the program should print an error message and exit immediately.
- Otherwise, after the user has entered all 1000 integers, the program should print the number of the user's integers that are less than 10.
- Duplicates still count – that is, if the user enters the integer 8 three times, that counts as 3 integers that are less than 10.