

**CISS240: Introduction to Programming
Quiz q0702**Name: YOUR EMAILScore:

This is a closed-book, no compiler, 5 minute quiz.

Q1. What is the output of the following code fragment? The repeating chunk of code appears 10 times; only 3 are shown below.

```
int s = 0;
int i = 0;

if (i % 2 == 1)
{
    s = s + i;
}
i = i + 1

if (i % 2 == 1)
{
    s = s + i;
}
i = i + 1

if (i % 2 == 1)
{
    s = s + i;
}
i = i + 1

... etc ...

std::cout << s << '\n';
```

ANSWER:

Q2. Complete the initialization of `i` so that the following program prints a random integer in the range of 50 to 60 (inclusive):

```
#include <iostream>
#include <cstdlib>
#include <ctime>

int main()
{
    srand((unsigned int) time(NULL));
    int i = _____;
    std::cout << i << '\n';
    return 0;
}
```

ANSWER:

```
int i = _____;
```

Q3. Complete the following program so that it prints a random double in the range of -1.0 to 3.0 (inclusive):

```
#include <iostream>
#include <cstdlib>
#include <ctime>

int main()
{
    srand((unsigned int) time(NULL));
    double x = _____;
    std::cout << x << '\n';
    return 0;
}
```

ANSWER:

```
double x = _____;
```

Q4. Complete the following program so that it prints a random integer from the following: 1, 2, 3, 4, 5, 20, 21, 22, 23, 24.

```
#include <iostream>
#include <cstdlib>
#include <ctime>

int main()
{
    srand((unsigned int) time(NULL));
    int i = _____;
    // write an if statement
    std::cout << i << '\n';
    return 0;
}
```

ANSWER:

```
int i =      ;  
if ()  
{  
}
```

(HINT: Generate a random integer 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 and put it in `i`. If `i` is 6, 7, 8, 9, 10, do something to `i`.)

INSTRUCTIONS

In the file `thispreamble.tex` look for

```
\renewcommand\AUTHOR{}
```

and enter your email address:

```
\renewcommand\AUTHOR{jdoe5@cougars.ccis.edu}
```

(This is not really necessary since alex will change that for you when you execute `make`.) In your bash shell, execute “`make`” to recompile `main.pdf`. Execute “`make v`” to view `main.pdf`.

Enter your answers in `main.tex`. In the bash shell, execute “`make`” to recompile `main.pdf`. Execute “`make v`” to view `main.pdf`.

For each question, you’ll see boxes for you to fill. For small boxes, if you see

```
1 + 1 = \answerbox{}
```

you do this:

```
1 + 1 = \answerbox{2}
```

`answerbox` will also appear in “true/false” and “multiple-choice” questions.

For longer answers that need typewriter font, if you see

```
Write a C++ statement that declares an integer variable name x.  
\begin{answercode}  
\end{answercode}
```

you do this:

```
Write a C++ statement that declares an integer variable name x.  
\begin{answercode}  
int x;  
\end{answercode}
```

`answercode` will appear in questions asking for code, algorithm, and program output. In this case, indentation and spacing is significant. For program output, I do look at spaces and newlines.

For long answers (not in typewriter font) if you see

```
What is the color of the sky?  
\begin{answerlong}  
\end{answerlong}
```

you can write

```
What is the color of the sky?  
\begin{answerlong}  
The color of the sky is blue.  
\end{answerlong}
```

A question that begins with “T or F or M” requires you to identify whether it is true or false, or meaningless. “Meaningless” means something’s wrong with the question and it is not well-defined. Something like “ $1 + 2 = 4$ ” is either true or false (of course it’s false). Something like “ $1+2 = 4?$ ” does not make sense.

When writing results of computations, make sure it’s simplified. For instance write 2 instead of $1 + 1$.

HIGHER LEVEL CLASSES.

For students beyond 245: You can put L^AT_EX commands in `answerlong`.

More examples of meaningless statements: Questions such as “Is $42 = 1+2$ true or false?” or “Is $42 = \{2\}^{\{3\}}$ true or false?” does not make sense. “Is $P(42) = \{42\}$ true or false?” is meaningless because $P(X)$ is only defined if X is a set. For “Is $1 + 2 + 3$ true or false?”, “ $1 + 2 + 3$ ” is well-defined but as a “numerical expression”, not as a “proposition”, i.e., it cannot be true or false. Therefore “Is $1 + 2 + 3$ true or false?” is also not a well-defined question.

More examples of simplification: When you write down sets, if the answer is $\{1\}$, do not write $\{1, 1\}$. And when the values can be ordered, write the elements of the set in ascending order. When writing polynomials, begin with the highest degree term.

When writing a counterexample, always write the simplest.