Quiz B   **Name:** _____

1) What will the variable `flag` equal here?

```
bool flag = (21 == 7*3);
```

2) What will the variable `flag` equal here?

```
bool flag = (6 + 2 < 15%5);
```

3) What's the difference between = and ==

4) What does `||` mean in an `if` statement?

5) What does `&&` mean in an `if` statement?

6) What will this print?

```
int cats = 6;
if (cats < 12)
{
     cout << "She has " << cats << " cats.\n";
}
else if (cats < 6)
{
     cout << "She is crazy, and has " << cats << " cats.\n";
}
else
{
     cout << "She is completely crazy with " << cats << " cats.\n";
}
```

7) What will this print? (hint see question 5, equivalence v assignment, this is a trick question)

```
int cats = 6;
if (cats = 1)
{
     cout << "She has one cat.\n";
}
else if (cats < 4)
{
     cout << "She has " << cats << " cats.\n";
}
else
{
     cout << "She is crazy, and has " << cats << " cats.\n";
}
```

8) The following code fragment is supposed to print whether the int variables value is odd or even, but it doesn't work. How can it be fixed?

```
switch (value%2)
{
    case 0:
        cout << "Even integer!" << endl;
    case 1:
        cout << "Odd integer!" << endl;
}
```

9) What will this print?

```
int i = 5;
while (i < 10)
{
    cout << "The number is " << i << endl;
    ++i;
}
```

10) What will this print?

```
for (int i = 0; i < 10; i += 2)
{
    cout << "The number is " << i << endl;
}
```

11) What will this print?

```
for (int i = 0; i < 6; ++i)
{
    cout << "The number is " << i << endl;
    if (i == 3)
    {
        break;
    }
}
```

12) What will this print?

```
for (int i = 0; i < 6; ++i)
{
    if (i == 3)
    {
        continue;
    }
    cout << "The number is " << i << endl;
}
```

13) Assuming the `#include <cmath>` library is in use, what will this print?

```
float fl = 66.6;
cout << floor(fl) << endl;
cout << round(fl) << endl;
```

14) Assuming the `#include <cmath>` library is in use, what will this print?

```
cout << sqrt(9) << endl;
```

15) What does this `foobar()` function do?

```
char foobar(char foo)
{
    if (foo > 'Z')
    {
        int bar = ('a' - 'A');
        foo = foo - bar;
    }

    return foo;
}
```

16) What does this `barfoo()` function do?

```
int barfoo(int foo, int bar)
{
    if (foo > bar)
    {
        return foo;
    }
    return bar;
}
```

17) What will this program print when run?

```cpp
int gThing = 5;

void myfunction(int j)
{
    j = j + gThing;
    cout << "j: " << j << endl;
}


int main()
{
    int i = 8;
    myfunction(i);
    cout << "i: " << i << endl;

    return 0;
}
```

18) **3 pts**. Write a C++ program that will print a <u>multiplication table</u> in the following form:

```
1
2    4
3    6    9
4    8    12   16
     .
     .
     .
10   20   30   ...
```

Your program must not-of course, consist entirely of `cout` statements. Use for loops.