Name(s):

1 const

1. In your own words, describe what the `const` keyword does.

2. Where are two places that you should use the `const` keyword?

3. Mark the following code. Circle the lines you think will cause compiler errors.

```
class Point {
public:
  Point(const int x, const int y)  {
    x_ = x;
    y_ = y;
  }

  int get_x() const { return x_; }

  int get_y() const {
    y_++;
    return y_;
  }

private:
  int x_;
  int y_;
};



void CreateSquare(const int y) {
  y = y * y;
}

void CreateCube(const int &y) {
  y = y * y * y;
}

void CreateQuadruple(int &y) {
  y = y * y * y * y;
}

int main() {
    int a = 10;
    const int b = 7;
    a++;
    b++;
}
```

Name(s):

2 Constructors

1. What is a constructor?

2. Write the equivalents of the following constructors using initialization lists.

```cpp
class Point {
public:
  // default constructor initializes Point at the origin
  // constructor 1
  Point() {
    x_ = 0;
    y_ = 0;
  }

  // constructor 2
  Point(const int x, const int y) {
    x_ = x;
    y_ = y;
  }

private:
  int x_;
  int y_;
};

class Library {
public:
  // constructor 3
  Library(const std::string name) {
    name_ = name;
    std::vector<Book> tmp;
    shelf_ = tmp;
  }

private:
  std::string name_;
  std::vector<Book> shelf_;  // Book is defined elsewhere in the code for us
};
```

3. Which of the constructors in the problem above are use in each of the following lines of code? How many instances of the object in question are created? (Or does the line of code cause an error?)

```cpp
Point p;

Point p2(1, 3);

Point p3[500];

Point p4(1, 2, 3);

Point * p5 = new Point();

Library lib;

Library * lib2 = new Library();

Library * lib3 = new Library("norlin");
```

Name(s): ____

3 `enum`

1. Create an enum class, `Suit` to represent the suit in a deck of cards.

2. Create a struct, `Card`, to represent a card in a deck of cards. You must use your `Suit` enum. You may use any other values that you choose.

3. Overload `operator==` in your `Card` struct.

4. Design a class, `Deck`. Write down the fields and methods that this class should have. Don't implement them!

4 Lecture 1 Review

1. What is a class? What is a struct? How are classes and structs different from one another?

2. Where can you access `public` fields? `private` fields? Are fields in a struct public or private?

Name(s):

3. What are the differences between the following three variable declarations? `int x = 2;`, `int * y = &x`, and `int &z = x;`

5 Writing nice code

1. Write down a new example of code that does not use good "boolean zen". Write down an example that does.

2. Write down a new example of a conditional statement that is not appropriated factored. Write down an example that is.

3. Write down a new example of a code that uses "magic numbers". Write down an example that does not.

4. How many tasks should a single function complete? What should you do if you find yourself writing a function that does not fit these guidelines? Give an example to demonstrate your answers.