

CS 1022C-001 - Spring 2014 - Exam 1
University of Cincinnati

There are 7 questions for a total of 100 points.

Name: _____

Instructions

- Please read through this entire exam very carefully before starting.
- This exam is closed notes and closed books.
- All work must be written on the exam pages in order to be graded. Any scrap paper used, must be the scrap paper provided during the exam period.
- For programming questions: Please be accurate with your C++ syntax: this includes appropriate use of braces, semicolons, and the proper use of upper/lowercase letters.
- No electronic devices may be used during the exam: this includes (but is not limited to) calculators, phones, tablets, and computers.
- You have 55 minutes to complete the exam.

DON'T PANIC!

| | | | | | | | | |
|---------------|----|----|----|----|----|----|----|-------|
| Question: | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Total |
| Points: | 10 | 21 | 15 | 12 | 12 | 30 | 0 | 100 |
| Bonus Points: | 0 | 0 | 0 | 0 | 0 | 0 | 15 | 15 |
| Score: | | | | | | | | |

Multiple Choice and True/False

1. Circle the **best** response.

2

(a) What command do you use to initialize your git repository from github.uc.edu?

- A. `git clone`
- B. `git pull`
- C. `git copy`
- D. `git fetch`

Solution: A - `git clone`

2

(b) What is the default type of parameter passing in C++?

- A. pass-by-value
- B. pass-by-reference
- C. pass-by-pointer

Solution: A - pass-by-value

2

(c) What is value of x and y after these lines execute?

```
1  int x = 5;
2  int y = 3 * ++x;
```

- A. x = 5, y = 15
- B. x = 6, y = 16
- C. x = 6, y = 18
- D. x = 5, y = 18

Solution: C - x incremented to 6 first, then multiplied by 3 for y = 18

2

(d) A for loop always executes at least one iteration.

- A. True
- B. False

Solution: False

2

(e) All pointers in a program are the same size.

- A. True
- B. False

Solution: True

Fill in the blank

2. Fill in the blank with the best answer

- 3 (a) The _____ member function is used to find out how many characters are in a C++ string object.

Solution: `length()`

- 3 (b) Arrays in C++ have a starting index of _____.

Solution: 0

- 3 (c) With this function signature `void myFun(int &x)`, x uses what type of parameter passing: _____.

Solution: pass-by-reference

- 3 (d) Defining multiple functions of the same name is called _____.

Solution: overloading

- 3 (e) _____ are special methods, that are used to initialize the state of an objects.

Solution: constructors

- 3 (f) _____ fields and methods on a class can be accessed anywhere you have access to an object.

Solution: public

- 3 (g) _____ is the act of hiding implementation details.

Solution: encapsulation

Code Analysis

- 15 3. Examine the following C++ code and identify the 5 syntax errors contained in the code

```
1 #include iostream
2
3 int main() {
4     cout << "Enter a character and a number";
5     cin >> ch, num;
6     if (ch == num);
7         cout << "The character entered matches the ascii code entered." << end;
8 }
```

Solution:

1. missing angle brackets on line 1
2. missing namespace declaration, using namespace std
3. variables on line 5 are not declared
4. , should be << on line 5
5. should be endl on line 7

The extra ; on 6 is not technically a syntax error, but rather a logic error.

Programming Questions

- 12 4. Leap years are years that are evenly divisible by 4, unless it is divisible by 100, unless it is also divisible by 400. For example 1900 was not a leap year, and 2000 was a leap year. Write a function called `isLeap` that accepts an integer and returns `true` if that number is a leap year and `false` if it is not.

Solution:

```
1 bool leap(int year) {  
2     return year % 4 == 0 && (year % 100 != 0 || year % 400 == 0);  
3 }
```

- 12 5. A liter is 0.264179 gallons. Write a function that takes in the number of liters of gasoline put into a car, and the number of miles that car was driven. The function will return the **miles per gallon** calculation. This often happens when visiting Canada, where gasoline is sold by the liter instead of the gallon.

Solution:

```
1 double mpg(double liters , double miles) {  
2     return miles / (liters * 0.264179);  
3 }
```

- 30 6. Write a class, called `Pixel` that represents a pixel as 3 integer values for red, green, and blue. The minimum allowed value for a single color is 0 and the maximum value allowed for any individual color is 255. Your constructor should take in integer values for red, green, and blue and make they they are constrained to the 0 to 255 range. If the value is < 0 , it should be made 0 and if it is > 255 , it should be made 255. Your class must contain the following methods.

- `int getRed()`
- `int getBlue()`
- `int getGreen()`
- `void brighten(double percent)`
- `void darken(double percent)`

The `brighten` method should increase the value of each color by the percent passed in, where 0.50 represents 50% increase. The `darken` method subtracts that amount. The result of the brighten/darken operation must be made to stay within 0 and 255, so that the pixel is still valid.

- 15 (bonus) 7. Add a method called `asHex` that returns the hex representation of the RGB values with 2 digits for example, if we were to declare a pixel object like this `Pixel p(122, 4, 250);`, the `asHex()` method would return a string containing `0x7a04fa`. Note that the green value, 4, is zero padded in the hex output. The leading `0x` must be added by your code as well.

Solution:

```
1 class Pixel {
2 public:
3     Pixel(int nr, int ng, int nb) {
4         r = normal(nr);
5         g = normal(ng);
6         b = normal(nb);
7     }
8
9     int getRed() {
10         return r;
11     }
12
13     int getGreen() {
14         return g;
15     }
16
17     int getBlue() {
18         return b;
19     }
20
21     void brighten(double percent) {
22         r = brighten(r, percent);
23         g = brighten(g, percent);
```

```
24     b = brighten(b, percent);
25 }
26
27 void darken(double percent) {
28     r = darken(r, percent);
29     g = darken(g, percent);
30     b = darken(b, percent);
31 }
32
33 string asHex() {
34     string s = "0x" + asString(r) + asString(g) + asString(b);
35     return s;
36 }
37
38 private:
39     int r;
40     int g;
41     int b;
42
43     static int darken(double val, double percent) {
44         return normal((int) (val - val * percent));
45     }
46
47     static int brighten(double val, double percent) {
48         return normal((int) (val + val * percent));
49     }
50
51     static string asString(int val) {
52         char buffer[2];
53         sprintf(buffer, "%2x", val);
54         string s(buffer);
55         return s;
56     }
57
58     static int normal(int val) {
59         if (val < 0) {
60             return 0;
61         } else if (val > 255) {
62             return 255;
63         }
64         return val;
65     }
66 };
```


Continued space for questions 6 and 7

Continued space for questions 6 and 7

SCRATCH PAPER