**CPSC 1020 FALL 2017**

**FINAL EXAM**

**This is a Closed Book exam. Please keep your notes and your computers closed. You have 2 hours to complete the Exam. There is a total of 65 points. Although the points total 65 your grade will reflect 100%. EX. A total score of 65 will result in 100%, a total score of 50 will result in a grade of 76.9%.**

1. **(3 points)**

Given the following declarations and initialization.

int x = 3;

int y = 6;

int \*ptr1 = &x;

int \*ptr2 = &y;

ptr2 = ptr1;

What is the output of the following program fragment

printf("%d %d - %d %d", x ,\*ptr1, y, \*ptr2);

1. 6 6 – 6 6
2. 3 3 – 3 3
3. 3 3 – 6 3
4. 3 6 – 3 3
5. **(6 points)**

**Part A.**

Consider the following struct:

typedef Pixel{

unsigned char red;

unsigned char green;

unsigned char blue;

}

In **C,** write the code to **dynamically allocate memory** for a 2D array of **Pixels** of size **width** and **height.** You can assume width and height have been declared and initialized.

Hint: Remember a Pixel is just a data type similar to the Circles struct from your lab.

**Part B.**

In **C++,** write the code to **dynamically allocate memory** for a 2D array of **Pixels** of size **width** and **height.** You can assume width and height have been declared and initialized.

Hint: Remember a Pixel is just a data type similar to the Circles struct from your lab.

1. **(4 Points – 2 each)**

Consider the following code:

Class A{

public:

//This function returns a + b

int add(int a, int b);

};

In class we discussed two examples of how to make the function “add” an inline function.

Provide the code --for both examples-- to make the function “add” inline.

Example 1:

Example 2:

1. **(3 points)**

Briefly, describe what happens, during compile time, when you have an inline function.

1. **(2 points)**

What is the output of the following code.

#include <stdio.h>

Output:

int f(int , int \*, int \*\*);

int main()

{

int c, \*b, \*\*a;

c = 5;

b = &c;

a = &b;

printf("%d \n", f(c, b, a));

return 0;

}

int f(int x, int \*py, int \*\*ppz)

{

int y, z;

\*\*ppz += 3;

z = \*\*ppz;

\*py += 2;

y = \*py;

x += 1;

return x + y + z;

}

**6. (4 Points)**

**What is the output of the following program?**

#include <stdio.h>

OUTPUT:

int main(){

int num1 = 5;

int num2 = 10;

int \*ptr = &num1;

int array[10] = {1,3,2,3,8,2,6,-1,0,9};

num2 += \*ptr;

printf("%d\n", num2);

num2 += \*ptr + 2;

printf("%d\n", num2);

printf("%d\n", \*ptr);

ptr = array;

ptr += 2;

printf("%d\n", \*ptr);

return 0;

}

1. **(5 Points)**

Consider the following class definition:

class Widget{

private:

double price;

int quantity;

public:

/\*Setters\*/

void setPrice(double);

void setQuantity(int);

/\*Getters\*/

double getPrice();

int getQuantity() ;

/\*The printInfo function prints the price of a widget and the quantity of a widget.\*/

void printInfo();

};

Write the implementation for the above class functions.

**8. (6 points )**

The following code has several mistakes. Circle and describe the mistakes in this code.

#include <iostream>

using namespace std;

class DumbBell{

int weight;

public:

void setWeight(int);

}

void setWeight(int w){

w = weight;

}

int main(){

DumbBell bar;

DumbBell.setWeight(200);

cout >> “The weight is “ >> weight << endl;

return 0;

}

**9 (6 points)**

Consider the program below: This is a program we spent extensive time, in class, analyzing.

#include <iostream>

**Explain the problem with this program.**

**Add the code to the program to fix this problem. Your solution must use the method we discussed in class. One line of code is all that is required here.**

using namespace std;

int main(){

char ch;

cout << "Type a character and press Enter: ";

cin >> ch;

cout << "You entered " << ch << endl;

cout << "This program has paused. Press Enter to continue.";

cin.get(ch);

cout << "It has paused a second time. Please press Enter again.";

ch = cin.get();

cout << "It has paused a third time. Please press Enter again.";

cin.get();

cout << "Thank you!\n";

return 0;

}

When I compile and run this program, I expect the program to do the following:

Type a character and press Enter: Y

You entered Y

This program has paused. Press Enter to continue.

It has paused a second time. Please press Enter again.

It has paused a third time. Please press Enter again.

Thank you!

However, this program actually ran as follows:

Type a character and press Enter: Y

You entered Y

This program has paused. Press Enter to continue.It has paused a second time. Please press Enter again.

It has paused a third time. Please press Enter again.

Thank you!

**10. (6 points)**

Rewrite the following C program in C++:

#include <stdio.h>

int main(int argc, char\* argv[]){

FILE\* input = fopen(argv[1], "r" );

int num;

while(fscanf(input, "%d", &num)== 1){

printf("%d\n", num);

}

fclose(input);

return 0;

}

Code goes here:

**11. (3 points)**

In class we discussed various types of constructors, one of which is a copy constructor.

**Briefly** explain why we should override the default copy constructor provided by C++.

**12. (4 points)**

Below is a declaration for a class called Array. You are to implement the copy constructor:

class Array

{

private:

int size;

int \*ptr;

public:

Array();

Array(int aSize);

Array(const Array &obj);

~Array();

int getSize()const;

static int getArrayCount();

Array &operator=(const Array &obj);

void print(int) const;

void setElement(int e, int value) ;

};

**Copy constructor** implementation goes here:

**13. (3 points)**

On the last page of this exam you will find the class declaration for Address, Date, and Person. Complete the Person default constructor using the **member initialization list technique**.

Person::Person()

**14. (3 points)**

In class we discussed something called the “RULE OF THREE”. Explain what this means:

**15. (3 Points)**

class MyString {

private:

char\* data;

int size;

public:

MyString ();

MyString (const char\*);

MyString (const MyString&);

~MyString ();

int len () const;

MyString& operator= (const MyString&);

MyString& operator= (const char\*);

1. friend MyString operator+ (const MyString&, const MyString&);
2. friend MyString operator+ (const MyString&, const char\*);
3. friend MyString operator+ (const char\*, const MyString&);

char operator[] (int ndx) const;

friend ostream& operator<<(ostream&, const MyString&);

};

The above MyString class is from your lab 12 assignment. Choose **one** of the three overloaded **operator+** functions and implement it:

**16. (6 points)**

In class we compared linked list, arrays, and vectors. As an example, linked list and vectors can grow and shrink as needed, arrays can not.

We discussed a specific example of when using a link list is better than using an array or vector. (There was a slide specific on this.)

**17. (1 points)**

With respect to a linked list. How can you tell if the list is empty?

class Address

{

private:

int house;

string street;

string city;

string state;

int zip;

public:

Address(int h, string str, string c, string st, int z);

};

class Date

{

private:

int month;

int day;

int year;

public:

Date();

Date(int m, int d, int y);

};

class Person

{

private:

string last;

string first;

string email;

Address address;

Date bday;

public:

Person();

Person(string l, string f, string e, int house, string street, string city, string state, int zip,

int month, int day, int year);

};