Name \_\_\_\_\_\_\_\_\_\_\_\_\_

Section # \_\_\_\_\_\_\_\_\_\_\_

CS120 Computer Science I

Second Exam

This is a closed note, closed book exam.

1. **(12 points)** Consider the following code:

void change1(int \*a){

\*a = \*a + 2;

}

void change2(int a){

a = a + 2;

}

void change3(int &a){

a = a + 2;

}

int main(){

int x = 10;

cout << x << endl;

change1(&x);

cout << x << endl;

change2(x);

cout << x << endl;

change3(x);

cout << x << endl;

}

What 4 numbers will this program print?

**2. (12 points)** Fill in the following table:

|  |  |  |
| --- | --- | --- |
| Binary | Base – 10 (decimal) | Hexadecimal |
| 1101 |  |  |
|  | 17 |  |
| 00101010 | 42 | 2a |

**3. (18 pts)** For the statements below circle T if the statement that follows is true, F if the statement is false.

T F a. The statement int data[100] creates an array with 100 values.

T F b. Hexadecimal numbers are written in base-16.

T F c. If a and b are both arrays of 20 integers then we can copy b into a with the statement a = b;

T F d. Array arguments to functions are always pass-by-value.

T F e. A class can have more than one constructor.

T F g. It is illegal to declare an array of objects.

T F f. A pointer stores the address of another variable.

T F h. The number 540 must be a decimal (base 10) number.

**4. (15 pts)** Consider the following code fragment:

int X = 5;

double f = 8.5;

double array[2] = {7.1,8.1};

double \*ptA, \*ptB;

ptA = array;

ptB = &f;

\*ptA = 22.5;

\*ptB = -12.5;

ptA = &ptB;

Fill in the table below showing what values would be stored where in memory *after* the code is run (one of them is done for you):

|  |  |  |
| --- | --- | --- |
| **Label** | **Value in memory** | **Actual memory address** |
| X | 5 | bbef0 |
| F |  | bbef8 |
| array[0] |  | bbf00 |
| array[1] |  | bbf08 |
| ptA |  | bbf10 |
| ptB |  | bbf18 |

**5. (23 pts)** Declare a class that stores information about rectangular solids. The name of the class should be *solid*. The class should have three private data members: height, width, and length, all of which should be doubles. The class should have four member functions you do need to write the code for these functions:

1. A public constructor that sets all three data members to the value 1.
2. A public function called volume() that returns the volume of a solid object.
3. A public function called set\_height()that takes a double as an argument and, if the argument is positive, sets the height variable to that value. If the height is negative it should print an error message.
4. A public function called is\_cube() that should return a 1 if the solid is a cube and a 0 otherwise.

**6. (10 points)** Write the prototype and the definition (i.e. the code) for a function called *backwards()* that takes two arguments. The first argument is an array of integers. The second argument is an integer that is the length of the array.

The function should print the array backwards.

**7. (10 points)** Write the prototype and the definition (i.e. the code) for a function called *reverse()* that takes two arguments. The first argument is an array of integers. The second argument is an integer that is the length of the array.

The function reverses the elements in the array. For example, if the array is initially:

Array1 = {1,2,3,4};

Then after the function is called it should be:

Array1 = {4,3,2,1};

**8. ()** Write a program that creates and prints a 10x10 multiplication table. The program should create a two-dimensional array and fill it with the values of a multiplication table:

1 2 3 …

2 4 6 …

3 6 …

…

Then the program should print the two-dimensional array.