**HOMEWORK 7 – Pointers**

Answer the following questions. **THIS ASSIGNMENT WILL BE DUE AT A DATE TO BE ANNOUNCED IN CLASS.** (points as noted – 100 points total)

1. A pointer variable stores a(n) memory address . (5 points)

2. The & operator is used to determine a variable’s address. (5 points)

3. What is the NULL pointer? What technically is its address? (10 points)

A null pointer is a pointer that points to nothing. Its address is technically zero.

4. Write the code to simply declare a float variable y and a pointer fPtr to y. Set the value of y equal to any value you wish via the pointer. (20 points)

#include <iostream>

using namespace std;

int main()

{

float y;

float\* fPtr;

fPtr = &y;

\*fPtr = 123.123;

cout << y;

}

5. Assume ptr is a pointer to int and holds the address 12000. On a system with four-byte integers, what address will be in ptr after the statement ptr += 10; ? (5 points)

The address will be 12040 because adding an integer, n, to a pointer increases the point, n, times the data type being pointed to.

6. Define a pointer charPtr to point to the string “Charmeleon.” What will the statement cout << charPtr+3; produce? Write the code to store the letter “l” in a char variable ch. (5 points)

#include <iostream>

using namespace std;

int main() {

string animal = "Charmeleon";

string\* charPtr = &animal;

cout << charPtr+3 << endl;

char ch = 'l';

cout << ch << endl;

return 0;

}

The statement cout << charPtr+3; will produce the address of the animal variable plus 3

7. Do the following (6 points each):

a. Define a structure Joe that contains an integer member intVar and pointer fPtr to type float.

struct Joe {

int intVar;

float\* fPtr;

};

b. Next, define an instance of the structure named Bob.

#include <iostream>

using namespace std;

struct Joe {

int intVar;

float\* fPtr;

};

int main()

{

Joe Bob;

return 0;

}

c. Write the code that sets the pointer of Bob to point to the variable y from Problem 4, then use this pointer to set the value of y equal to 11.17.

#include <iostream>

using namespace std;

struct Joe {

int intVar;

float\* fPtr;

};

int main()

{

Joe Bob;

float y = 0.0;

Bob.fPtr = &y;

\*Bob.fPtr = 11.17;

return 0;

}

d. Define a pointer sPtr to this structure and initialize it to point to Bob.

#include <iostream>

using namespace std;

struct Joe {

int intVar;

float\* fPtr;

};

int main()

{

Joe Bob;

float y = 0.0;

Bob.fPtr = &y;

\*Bob.fPtr = 11.17;

Joe\* sPtr = &Bob;

return 0;

}

e. Use the pointer to set intVar equal to 177 and to reset y to 863.33.

#include <iostream>

using namespace std;

int main()

{

struct Joe {

int intVar;

float\* fPtr;

};

Joe Bob;

float y = 0.0;

Bob.fPtr = &y;

\*Bob.fPtr = 11.17;

Joe\* sPtr = &Bob;

sPtr->intVar = 177;

\*sPtr->fPtr = 863.33;

cout << "Bob.intVar: " << Bob.intVar << endl;

cout << "y: " << y << endl;

return 0;

}

8. Use the new operator to dynamically allocate an integer array of size n (to be entered by the user via a cin prompt). (20 points)

#include <iostream>

using namespace std;

int main()

{

int n;

cout << "Enter the size of the array." << endl;

cin >> n;

int\* arr = new int[n];

delete[] arr;

cout << "The size of the array is "<< n << endl;

return 0;

}