

FINAL CMPSC 16, PROBLEM SOLVING WITH COMPUTERS, SUMMER 2016

NAME:

DATE:

Problem 1.(10 points.) What programming language did we learn in this class?

C++

Problem 2.(10 points.) If `int x = 12`, what will be the value of `!x`?

0

Problem 3.(10 points.) What will this expression evaluate to (true or false)?

```
!(true && false) && (!true || !false) || (!true && !false)
```

True

Problem 4.(10 points.) What is the difference between `char[]` and `string`?

char[] is simply an array of characters whereas string is a class, which has far more functionality.

Problem 5.(10 points.) What will this program output?

```
#include <iostream>

using namespace std;

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

    string str = "calfs_in_16";
    int array[] = {0,4,9,10,5,3,6,7,1,2};

    for (int i: array) {
        cout << str[i];
    }

    return 0;
}
```

cs16 final

Problem 6.(10 points.) What will this program output?

```
#include <iostream>

using namespace std;

void do_something(int i) {
    if (i == 0) {
        return;
    }
    cout << i%10;
    do_something(i/10);
}

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

    do_something(123456);

    return 0;
}
```

654321

Problem 7 [Vectors].(10 points.) What should this program do? Where is the bug, and what effect will it have? Fix it.

```
#include <iostream>
#include <vector>

using namespace std;

void do_something(string& s) {
    for (int i = 0; i < s.length(); i++) {
        s[i] = toupper(s[i]);
    }
}

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

    vector<string> inputs(0);
    int idx = 0;

    cout << "Give me strings:" << endl;
    while (1) {
        string tmp;
        getline(cin, tmp);

        if (tmp == "")
            break;

        inputs[idx++] = tmp;
    }

    for (int i = 0; i < inputs.size(); i++) {
        do_something(inputs[i]);
    }

    for (string s: inputs) {
        cout << s << endl;
    }
}
```

This program will take a list of strings as input (terminating when the user inputs an empty line), and then output the given strings in all UPPERCASE. The bug is that the vector is initialized to 0, and then indexed directly, which will result in memory corruption. We should use `push_back` instead.

Problem 8 [Char Magic].(10 points.) What should this program do? Where is the bug, and what effect will it have? Fix it.

```
#include <iostream>

using namespace std;

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

    // Note: 'A' = 0x41 = 61
    // and 'Z' = 0x5A = 90
    int inputs['Z'-'A'+1] = {0};

    cout << "Give me letters:" << endl;
    char c;
    do {
        cin.get(c);
        c = toupper(c);
        if (c >= 'A' && c <= 'Z') {
            ++inputs[c];
        }
    } while (c != '\n');

    for(int i = 'A'; i <= 'Z'; i++) {
        if (inputs[i-'A']) {
            cout << i << " :_" << inputs[i-'A'] << endl;
        }
    }
}
```

This program will take an input of characters and output their frequencies. The bug is that when we increment the count for the character (i.e., `++inputs[c];`), we do not subtract 'A', which will result us updating random values in memory and not our actual array. The fix: `++inputs[c-'A'];`

Problem 9 [Pointer Magic].(10 points.) What will this program do (brief description)? [i.e., What will **List 1** and **List 2** output with respect to the input numbers?]

```
#include <iostream>
#include <vector>

using namespace std;

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

    vector <int *> v1, v2;
    int num, tmp;

    cout << "How_many_numbers?" << endl;
    cin >> tmp;

    int *nums = new int[tmp]();

    cout << "Give_me_numbers:" << endl;
    for (int i = 0; i < tmp; i++) {
        cin >> nums[i];
    }

    cout << "Give_me_a_number:" << endl;
    cin >> num;

    for (int i = 0; i < tmp; i++) {
        if (nums[i] <= num)
            v1.push_back(nums+i);
        else
            v2.push_back(nums+i);
    }

    cout << "List_1:_";
    for (int *i: v1)
        cout << *i << "_";
    cout << endl;

    cout << "List_2:_";
    for (int *i: v2)
        cout << *i << "_";
    cout << endl;

    return 0;
}
```

It will read in a list of numbers and split them into two lists where List 1 contains all of the numbers that are less than or equal to the input number and List 2 contains those numbers that are greater than the input number.

Problem 10 [File input].(10 points.) What will this program do (brief description)?

```
#include <iostream>
#include <string>
#include <fstream>
#include <vector>
#include <cmath>
using namespace std;
struct Location {
    int posx;
    int posy;
    string name;
};
struct ReturnLoc {
    int loc1 = 0;
    int loc2 = 0;
    double d = _DBL_MAX_; // Maximum possible value for a double
};
void do_something(int idx, vector<Location> v, ReturnLoc& r) {
    if (idx >= v.size())
        return;
    for (int i = idx+1; i < v.size(); i++) {
        double d = sqrt(pow(v[i].posy - v[idx].posy, 2) +
            pow(v[i].posx - v[idx].posx, 2));
        if (d < r.d) {
            r.loc1 = idx;
            r.loc2 = i;
            r.d = d;
        }
    }
    do_something(idx+1,v,r);
}
int main(int argc, const char * argv[]) {
    vector<Location> locations;
    string filename;
    ifstream in_stream;
    ReturnLoc rtn;

    cout << "Input_file:" << endl;
    cin >> filename;

    in_stream.open(filename);
    while (!in_stream.eof()) {
        Location l;
        in_stream >> l.posx;
        in_stream >> l.posy;
        if (in_stream >> l.name) // False at end of file
            locations.push_back(l);
    }
    in_stream.close();

    do_something(0, locations, rtn);

    cout << "Locations:_ " << locations[rtn.loc1].name << "_and_";
    cout << locations[rtn.loc2].name << endl;
    cout << "Result:_ " << rtn.d << endl;
}
```

It will take in a list of “Locations,” denoted by x-y coordinates and a name (x y name) on each line, and return the closest pair.

