Homework 1

Object Definition and Debugging

C++ - Introduction

University of Washington Outreach Program

Name\_\_\_\_\_

1. As a first step into the world of design let's work with a cell phone. Your company, *Personal Communications Unlimited*, Ltd. has decided to develop the next generation mobile device to be ready by the holidays. You've just been assigned to lead the task. You begin by thinking about what is important to your customers and how they might want use it. Based upon your thoughts, you propose an initial set of features and capabilities for the device.

For each feature or capability that you select, write one or two sentences describing that feature, in detail, from the user's point of view. Be sure to indicate why you are including it, what its purpose is, and the benefit(s) to the user.

Also think about the feature from the negative perspective. That is, if you did not include the feature, how would that affect the usefulness, effectiveness, and attractiveness of your product?

Finally, think about the effect on the cell phone if that feature were to fail.

At this stage, do not think about how you might implement it. That's for later. For now, focus on the product from the outside. Focus on what behavior that the user might expect or want to see.

2. Consider the following program that Clyde P. Barkwasser, a new student to C++, wrote to practice working with character arrays and stream I/O.

Working with your debugger, please determine if the program accomplishes what Clyde intended to do. If there are any errors, please identify the line or lines that may be in error, identify what the error is, and how you found it with your debugger. Please make the appropriate correction(s) and show that your proposed modification fixes the original problem.

```
#include <iostream>
using namespace std;
// this is a simple routine that demonstrates how to fill an array of characters
void main(void)
{
    char myArray[5];
                                          // declare a character array
    for (int i = 0; i \le 5; i++)
                                          // fill array with characters
    {
        // fill with the ascii characters A..F
        // 65 is the ascii value for A
        myArray[i]= 65+i;
    }
    for (i = 0; i <= 5; i++)
                                               // display the array
    {
        cout << myArray[i];
    }
    cout << endl;
    return;
}
```

3. Clyde's sister, Brunelda Raisondetre, also in the class, is practicing using function prototypes and passing variables to functions. She's written the following program. What do you have to say to Brunelda. Use your debugger to illustrate your analysis. Can you correct her error(s) using things we may have covered in class? Please make the appropriate correction(s) and show that your proposed modification fixes the original problem.

Please note that you are changing what the pointers refer to, not the contents of the containers to which they are referring.

```
#include <iostream>
using namespace std;
void swapPointers (int* aPtr0, int* aPtr1);
void main(void)
{
    // declare a couple of arrays
    int myArray0[] = \{1, 2, 3\};
    int myArray1[] = \{4, 5, 6\};
    // swap them
    swapPointers (myArray0, myArray1);
    // test to make sure it worked
    for (int i = 0; i < 3; i++)
    {
        cout << myArray0[i] << " " << myArray1[i] << endl;
    }
    return;
}
// this routine interchanges the pointers to two arrays
// inputs:
                pointers to two arrays
// outputs:
                none
// function:
                accept pointers to two arrays. Interchange them such that the first
//
                pointer points to the second array and the second pointer to the first.
void swapPointers (int* aPtr0, int* aPtr1)
{
    // swap the pointers
    int* tempPtr = aPtr0;
    aPtr0 = aPtr1;
    aPtr1 = tempPtr;
    // test it to make sure it works
    for (int i = 0; i < 3; i++)
    {
        cout << aPtr0[i] << " " << aPtr1[i] << endl;
    }
    return;
}
```

4. By some strange twist of fate, Clyde and Brunelda's cousin, Ascot B. Pompousse is also in the same class. Being a legend in his own mind, Ascot puts together the following program to illustrate how much better he is at C++ than his cousins.

Again using your debugger, give poor Ascot a lesson in proper code design. Be sure to explain and correct his error(s). Please show that your proposed modification fixes the original problem.

```
#include <iostream>
using namespace std;
// get data from the user
void getData(int* aValuePtr);
void main (void)
{
    // declare a shared variable and a pointer to it
    int myValue:
    int* myPtr = &myValue;
    // get data from the user
    getData(myPtr);
    // display the data
    cout << *myPtr << endl;
}
// prompt the user for some data and return it through a shared
// variable pointed to by valuePtr
// inputs: pointer to a container in which to place the data
// outputs: none
// function: the routine accepts a pointer to a container in which to store data from a user,
// it prompts for the data, accepts the data, displays it, and returns
void getData(int* valuePtr)
{
    // declare a temp place to store the data
    int tempValue;
    // let valuePtr point to it
    valuePtr = &tempValue;
    // prompt for data
    cout << "Please enter a value" << endl;
    // get the data
    cin >> *valuePtr;
    // display its value
    cout << *valuePtr << endl;
    return;
}
```