EECE 230 Introduction to Programming, Sections 1 and 2
Quiz I

March 28, 2011

- The duration of this exam is 3 hours.
- It consists of 4 problems
- The exam is open book. You can use also all the material on Moodle: lecture notes, programming assignments, and solutions, etc. You are **NO**T allowed to use the **web** (**imail** included). You are not allowed to use **USB's** or files previously stored in your **account**.
- If you violate the above rules or if you communicate with a person other than the exam proctors during the exam, you will immediately get zero and you will be referred to the appropriate disciplinary committee.
- Active cell phones and any other unauthorized electronic devices are absolutely not allowed in the exam rooms. They should be turned off and put away.
- Plan your time wisely. Do not spend too much time on any one problem. Read through all of them first and attack them in the order that allows you to make the most progress.
- Submit your solutions each part in a separate file as indicated in the booklet. Include your name and ID number in each file. Submit the files online in a single zip file called *yourLastName.yourFirstName.zip*.
- Good luck!

Problem 1 (20 points). Closest pair of points

In this problem, you are not allowed to use for/while loops or arrays.

Write a program which asks the users to enter three planar points: (x_1, y_1), (x_2, y_2), and (x_3, y_3), where $x_1, y_1, x_2, y_2, x_3, y_3$ are real numbers. Your program is supposed to find the *distance between the closest pair of the three input points*.

Note that the distance between two planar points (x, y) and (x', y') is given by $d((x, y), (x', y')) = \sqrt{(x - x')^2 + (y - y')^2}$.

To find the square root, use the function *pow* in the *cmath* library.

Sample input/output:

```
Enter the (x,y) coordinates of the first point:1.2 3.0
Enter the (x,y) coordinates of the second point:4.1  -6
Enter the (x,y) coordinates of the third point:2  7

The distance between the closest pair is: 4.07922
```

Check: $d((1.2, 3.0), (4.1, -6)) = 9.45569$, $d((1.2, 3.0), (2, 7)) = 4.07922$, and $d((4.1, -6), (2, 7)) = 13.1685$. Thus the minimum is 4.07922.

Submit your solution in a file called Prob1.cpp including your name and ID number.

Problem 2 (25 points). Sepcial integers

Write a program which given an *integer* n, checks whether or not there exists *integers* x and y, where $0 \leq x \leq 15$ and $0 \leq y \leq 15$, such that $n = 2^x + 5 \times 3^y$. If the answer is YES, your program is supposed to find such x and y.

Sample input/output:

```
Enter  and integer:1
NO
Enter  and integer:5
NO
Enter  and integer:6
YES: 6=2^0+5*3^0
Enter  and integer:7
YES: 7=2^1+5*3^0
Enter  and integer:8
NO
Enter  and integer:17
YES: 17=2^1+5*3^1
Enter  and integer:18
NO
Enter  and integer:53
YES: 53=2^3+5*3^2
Enter  and integer:54
NO
Enter  and integer:885863
YES: 885863=2^7+5*3^11
Enter  and integer:923232
NO
```

Submit your solution in a file called Prob2.cpp including your name and ID number.

Problem 3 (25 points). File copy

Write a program which asks the user to enter the name of an input text file and the name of a output text file. Your program should copy the content of the given input file into the output file. That is, it is supposed to create an output file whose content is a copy of the given input file.

If there is no input file with the given name, your program is supposed to print the error message "Cannot open input file!" and exit.

If the output file name is the same as the input file name, it is supposed to give the error message "Self copy!" and exit.

You can assume the names of the input and output files do not contain white spaces (thus you can safely read them using the extraction operator).

Submit your solution in a file called Prob3.cpp including your name and ID number.

Problem 4 (30 points). Smallest cut in array to make it sorted

Write a program which given a list of integers, finds the smallest sublist of the input list whose removal makes the input list sorted in nondecreasing order.

Assume the input list end is indicated by the sentinel -999. Store the list in an array $A[0 \dots n]$, where n is the number of integers in the list.

If A is sorted in nondecreasing order, your program is supposed to say "Array sorted: no cut needed" and exit. Otherwise, you are asked to find two indicies $0 \le i \le j \le n-1$ such that:

- If you remove the subarray $A[i \dots j]$ form the array $A[0 \dots n-1]$ you get a sorted array. That is,

  $A[0] \le A[1] \le \dots \le A[i-1] \le A[j+1] \le A[j+2] \le \dots \le A[n-1]$, if $i \ne 0$ and $j \ne n-1$

  $A[j+1] \le A[j+2] \le \dots \le A[n-1]$, if $i = 0$

  $A[0] \le A[1] \le \dots \le A[i-1]$, if $j = n-1$

- The length $l = j - i + 1$ of the subarray $A[i \dots j]$ is as small as possible

Note that $l \le n-1$ since you can always keep the first element or the last element.

Examples. The smallest cut is underlined in the examples below.

$A = \langle 1, 2, 3, 4, \underline{\mathbf{1}} \rangle$ $(i = 4,\ j = 4,\ l = 1)$

$A = \langle \underline{\mathbf{10}}, 1, 2, 3 \rangle$ $(i = 0,\ j = 0,\ l = 1)$

$A = \langle 1, 2, 3, 10, 11, \underline{\mathbf{4}, \mathbf{5}} \rangle$ $(i = 5,\ j = 6,\ l = 2)$.

$A = \langle 1, \underline{\mathbf{2}, \mathbf{3}}, 1, 1, 4, 5, 6 \rangle$ $(i = 1,\ j = 2,\ l = 2)$.

$A = \langle 1, 2, 3, \underline{\mathbf{7}, \mathbf{0}, \mathbf{1}, \mathbf{9}, \mathbf{1}}, 4, 5, 20 \rangle$ $(i = 3,\ j = 7,\ l = 5)$.

$A = \langle 1, 2, 3, 4 \rangle$ (Array sorted: no cut needed)

Any correct solution is worth 20/30.

If you solve it using doubly-nested loops and without using triply-nested loops (i.e., in quadratic time), you will get full grade.

If solve it without nested loops (i.e., in linear time), you will get 10 extra bonus points (this is difficult ... do not attempt to do it unless you are really sure that everything else in the exam is correct).

Submit your solution in a file called Prob4.cpp including your name and ID number.