EECE 231 — Introduction to Programming Using C++ and MATLAB, Sections 1,2,6,7, and 8
Quiz I

Oct 11, 2014

- The duration of this exam is 2 and a half hours.

  Keep in mind that you need around **10 minutes** at the end of the duration of the exam to **upload** your answers to the moodle website. It is your responsibility to make sure your files are correctly uploaded.

- The exam consists of 3 problems for 105 points.
- You can use all the material available on moodle.
- You are **NOT** allowed to use the **web**. You are not allowed to use **USB's** or files previously stored on your machine.
- If you get caught violating 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.
- The problems are of varying difficulty. Below is rough ordering estimate of the problems according to difficulties:
  - Problem 1 Parts (a) and (b), Problem 2 Part (a), Problem 3 Part (a)
  - Problem 1 Part (c), Problem 3 Part (b)
  - Problem 2 Part (b)

  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.
- Good luck!

Problem 1 (45 points). Rectangles

Write a program which prompts the user to enter the coordinates of the lower left corner of a rectangle (p_x, p_y), its width w, and its height h. The data type of the input parameters p_x, p_y, w, and h is *double*. After reading the input, you are asked to do the following.

a) **(15 points)** Compute and output the coordinates of the other three corners of the rectangle.

b) **(15 points)** Prompt the user to enter a point (x, y) and then check whether or not it is inside the rectangle.

c) **(15 points)** Print all the points with *integer* coordinates inside the rectangle. See the example below.

(*Note:* You might need to compute the ceiling of a real number. The ceiling c of a real number x is the smallest integer larger than or equal to x, for instance the ceiling of 3.1 is 4. In C++, the ceiling can be computed as follows:

```
int c = x; // Initialize c to  the floor of x (integer part of x)
  // Assigning a double to an integer gives a warning which you can either
  // ignore or avoid by using :
  // int c = static_cast<int>(x); //  similar to the above but  warning-free

if(c<x) c = c+1; //Compute the ceiling
```

Below is a sample input/output:

```
Enter p_x and p_y: 1.1 2.3
Enter width: 5.0
Enter height: 4.0

a) Coordinates of the other 3 corners: (6.1,2.3), (1.1,6.3), (6.1,6.3),

b) Enter x and y: 2 4
YES: (2,4) is inside the rectangle

c) Points with integer coordinates inside the rectangle:
(2,3) (2,4) (2,5) (2,6)
(3,3) (3,4) (3,5) (3,6)
(4,3) (4,4) (4,5) (4,6)
(5,3) (5,4) (5,5) (5,6)
(6,3) (6,4) (6,5) (6,6)

Press any key to continue . . .
```

Any correct solution is worth full grade. Submit your solution in a file called Prob1.cpp including your name and ID number.

Problem 2 (30 points). Divisibility by consecutive integers

a) **(15 points) Divisibility by two consecutive integers larger than one.** Write a program which prompts the user to enter a positive integer n and checks whether or not n is divisible by two consecutive integers larger than one. That is, check whether or not there exists an integer $d \geq 2$ such that both d and $d+1$ divide n. In addition to a YES/NO answer, your program should print (any) two consecutive integer divisors of n (larger than one) if the answer is YES. If n is not positive, your program should display the error message "Number not positive" and exit.

Examples:

- 24 is divisible by the consecutive integers 2 and 3.
- 100 is divisible by the consecutive integers 4 and 5.
- 112 is divisible by the consecutive integers 7 and 8.
- None of the following integers are divisible by consecutive integers (larger than one):

$$2, 3, 4, 10, 25, 44, 99$$

Any correct solution is worth 10 points. Faster solutions are worth more points.

Submit your solution in a file called Prob2a.cpp including your name and ID number.

b) **(15 points) Exact product of consecutive integers.** Write a program which prompts the user to enter a positive integer n and checks whether or not n is the **exact product** of at least two consecutive integers larger than one. That is, check whether or not there exists integers $b > a \geq 2$ such that $n = a(a+1)(a+2)\ldots(b-1)b$. In this part a YES/NO answer is enough. As above if n is not positive, your program should display the error message "Number not positive" and exit.

Examples:

- 6 is an exact product of the consecutive integers (2×3)
- 20 is an exact exact product of the consecutive integers (4×5)
- 24 is an exact product of the consecutive integers $(2 \times 3 \times 4)$
- 30 is an exact product of the consecutive integers (5×6)
- 210 is an exact product of the consecutive integers $(5 \times 6 \times 7)$
- None of the following integers is an exact product of consecutive integers (larger than one):

$$2, 3, 4, 5, 7, \ \ldots, 19, 21, 22, 23, 25, \ldots, 29, 100, 112$$

Note that although 100 is divisible by the consecutive integers 4 and 5, it is not the the exact product of consecutive integers. The same holds for 112.

Any correct solution is worth 10 points. Faster solutions are worth more points (*Hint:* to get full credit, do it without nested loops). Detailed comments and explanations of your code are worth partial credit.

Submit your solution in a file called Prob2b.cpp including your name and ID number.

Problem 3 (30 points). Sum of distances

Write a program which prompts the user to enter an integer n followed by a list of n real numbers. First, your program is supposed to store the user input in an array $A[0 \dots n-1]$. Then you are asked to do the following:

a) **(15 points) Sum of distances from the first point.** Compute and print the sum of distances between all the numbers in the list and the first element, i.e., $|A[1] - A[0]| + |A[2] - A[0]| + \dots + |A[n-1] - A[0]|$.

   For instance, if $A = \{1.5, 10.1, 3, 2.2, 6, 7\}$, then the answer is $|10.1 - 1.5| + |3 - 1.5| + |2.2 - 1.5| + |6 - 1.5| + |7 - 1.5| = 20.8$.

   Any correct solution is worth full grade.

b) **(15 points) Minimize wire length.** Imagine that the numbers in A represent the coordinates of points on a line. You are interested in connecting all the points in A to a single point in A by wires. You want to choose the point in A which minimizes the total wire length. That is, find the index i_{opt} of the element $A[i_{opt}]$ in array A such that the sum of all distances between all the the numbers in A and $A[i_{opt}]$ is minimal. In other words, you are asked to find the index i_{opt}, where $0 \le i_{opt} \le n-1$, such that $|A[0] - A[i_{opt}]| + |A[1] - A[i_{opt}]| + \dots + |A[n-1] - A[i_{opt}]|$ is minimal.

   For instance, if $A = \{1.5, 10.1, 3, 2.2, 6, 7\}$, then the answer is $i_{opt} = 2$ or $i_{opt} = 4$ since:

   - For $i = 0$, we get $|10.1 - 1.5| + |3 - 1.5| + |2.2 - 1.5| + |6 - 1.5| + |7 - 1.5| = 20.8$
   - For $i = 1$, we get $|1.5 - 10.1| + |3 - 10.1| + |2.2 - 10.1| + |6 - 10.1| + |7 - 10.1| = 30.8$
   - For $i = 2$, we get $|1.5 - 3| + |10.1 - 3| + |2.2 - 3| + |6 - 3| + |7 - 3| = \mathbf{16.4}$ (minimal)
   - For $i = 3$, we get $|1.5 - 2.2| + |10.1 - 2.2| + |3 - 2.2| + |6 - 2.2| + |7 - 2.2| = 18$
   - For $i = 4$, we get $|1.5 - 6| + |10.1 - 6| + |3 - 6| + |2.2 - 6| + |7 - 6| = \mathbf{16.4}$ (minimal)
   - For $i = 5$, we get $|1.5 - 7| + |10.1 - 7| + |3 - 7| + |2.2 - 7| + |6 - 7| = 18.4$

   Any correct solution is worth full grade.

   Detailed comments and explanations of your code are worth partial credit.

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