Program cop2270pgm2f

Make sure your program is properly documented and aligned uniformly, looking professionally, I will take points off if it not.

Purpose: Implement functions, if, else if, else, for, printf, primitive types, String class with several functions, single dimension array.

1 - In main() (Worth 2 points)
- Create the following variables:
  int i = 7;
  int j = 9;
  double x = 72.5;
  double y = -0.34;

- Passing the previous variables, call the following functions:
  processAbsoluteValues
  processRoundValues
  processCeilingValues
  processFlooringValues
  processMinimumValues
  processMaximumValues
  processTrigFunctionsValues (cos, sin, tan)
  processExponentialValues
  processLogValues
  processPowerValues
  processSquareRootsValues
  processRandomValues

- Create a single dimension array containing ONE index, which will have ONE string with your full name as follows:

  "first name $ middle name % last name and ONE space"
  i.e. "George & Washington % Rodriguez"
  Note: if you do not have a middle name use MN.

2 - (Worth 2 points) Create the previous functions, and in their corresponding functions, using the data received implement the following Math functions:
   abs, round, ceil, floor, min, max, cos, sin, tan, exp, log, pow, sqr, random

3 - Create a method called myName (Worth 2 points).
   Using the for loop, if, else if, else, ||, &&, printf, and any other command you want that we have learned in class,
   - Examine each character of your name and determine if it is:
     - a space, a vowel (a,e,i,o,u,A,E,I,O,U), a consonant (b,w,g, etc), the symbol % or the symbol $, and using the current value of x in your for loop print:

     character [x] located at position x is a consonant or
     character [x] located at position x is a vowel or
     character [x] located at position x is a space or
     character [x] located at position x is a symbol

     ie: My name is [George $ Washington % Rodriguez]

     character [G] located at position 0 is a consonant
     character [e] located at position 1 is a vowel
     character [o] located at position 2 is a vowel
     character [r] located at position 3 is a consonant
     character [g] located at position 4 is a consonant
     character [e] located at position 5 is a vowel
     character [ ] located at position 6 is a space
4 - Create a method called pyramid (Worth 2 points).
Using a for loop, display to the screen, the string containing your name so that each loop will NOT contain the first and the last character from the previous line, with the length of the string being printed, and the string surrounded by square brackets []

NOTE: Every method dealing with your name, MUST be called from main, PASSING your name to it.

i.e: 32 [George $ Washington % Rodriguez ]
30 [eorge $ Washington % Rodriguez]
28 [orge $ Washington % Rodri]
26 [rge $ Washington % Rodrig]
24 [ge $ Washington % Rodrig]
22 [e $ Washington % Rodri]
20 [ $ Washington % Rodr]
18 [$ Washington % Rod]
16 [ Washington % Ro]
14 [Washington % R]
12 [ashington %]
10 [shington %]
 8 [hington ]
 6 [ington]
 4 [ngto]
 2 [gt]
 0 []

5 - In a method called parsing do the following:

NOTE: Every method dealing with your name, MUST be called from main, PASSING your name to it.

- Print your name in upper case letters.
- Print your name in lower case letters.
- Print your name taking all spaces out.
- Print your name with all vowels in upper case, and all consonants in lower case.
- Print your name backwards.
- Print your name in ASCII values.

Turn in the source code on paper, and email me the source code.
Make sure the program is properly documented and aligned uniformly, looking professionally.