CSCI-581 **Homework 6 Linear regression.**

This is an individual programming assignment.

Given multidimensional data, input vectors $\boldsymbol{x_j} = [x_1, x_2]$ and output labels y_j, find the weights w_0, w_1, w_2 such that $h = w_0 + w_1 x_1 + w_2 x_2$ best fits the given data. Use the gradient descent update learning rule:

$$w_i = w_i + \alpha \sum_{j=1}^{n} x_{ji} \left(y_j - h(x_j) \right),$$

To solve this problem, write a program and run it with different number of update iterations. Initialize all weights to 1, then repeat the following two steps *N* times, where *N* is the given number of iterations:

Step 1: for all input vectors x_j, find

$$h(x_j) = w_0 x_{j0} + w_1 x_{j1} + w_2 x_{j2}$$

Step 2: update w_0, w_1, w_2 according to this learning rule:

$$w_i = w_i + \alpha \sum_{j=1}^{n} x_{ji} \left(y_j - h(x_j) \right),$$

Example:

$\boldsymbol{x_1} = [1, \ 2] \rightarrow -13$
$\boldsymbol{x_2} = [2, -1] \rightarrow \ -4$
$\boldsymbol{x_3} = [5, -2] \rightarrow \ \ 5$
$\boldsymbol{x_4} = [3, -1] \rightarrow \ \ 0$
$\boldsymbol{x_5} = [8, \ 2] \rightarrow \ \ 1$

Input to your program:

Line 1: M, integer, the number of input vectors (in the Example above, there are 5 input vectors).

Line 2: D, integer, the number of attributes for each x (in the Example above, D equals to 2, because each vector x has 2 attributes).

Line 3: N, integer, the total number of iterations that your program must run.

Line 4: α, double, use this α to update weights

Lines (rest of lines): total of M lines, each having (D + 1) integer values (D attributes and the last value is y-label); Do not forget to add a dummy attribute 1 for each x-vector.

Input Sample for the given Example:

```
5
2
100
0.01
1       2       -13
2       -1      -4
5       -2      5
3       -1      0
8       2       1
```

After your program reads in this input, it should add a dummy attribute to each x-vector and initialize corresponding y-labels as shown below:

$\boldsymbol{x_1} = [1, 1, \ \ 2]$	$\boldsymbol{y_1} = -13$
$\boldsymbol{x_2} = [1, 2, -1]$	$\boldsymbol{y_2} = \ -4$
$\boldsymbol{x_3} = [1, 5, -2]$	$\boldsymbol{y_3} = \ \ 5$
$\boldsymbol{x_4} = [1, 3, -1]$	$\boldsymbol{y_4} = \ \ 0$
$\boldsymbol{x_5} = [1, 8, \ \ 2]$	$\boldsymbol{y_5} = \ \ 1$

Output of your program:

Output the values w_i after *N* iterations (output ***endl*** after each value). For our example, you'll need to output the following values:

w_0
w_1
w_2

Output Sample for our Example:

```
-7.223076651005

1.513769015099

-2.455534721592
```

Use ***double*** variables for weights.

Use the following formatting syntax to output the results (include <iomanip> directive):

cout << std::fixed << std::showpoint << std::setprecision(12) << result << endl;

Submission: Submit the file named ***hw6.cpp*** to ***hw6*** on ***turnin***.