

General Assignment Programming Project 4EE11

The scores of students that took part in an exam can be evaluated by a small program. The grades are listed in a test file that is read by the program, which then calculates the average grade, the standard deviation and prints a histogram.

The code consists of a main program (grades.c), a library of functions (mylib.c and mylib.h) and a test program (test.c). The test program is designed to test the functions in the mylib library.

Unfortunately, the program has not yet been completed. It contains a number of syntax and semantic errors. In addition, some of the routines have not yet been implemented.

1. Before compiling and running the program, read the code carefully and see if you can identify the errors. Ten errors have intentionally been added to the code. List the errors in a table as shown below.

Next, you may try to compile the code. Add the newly found errors in the table and mention the compiler messages (warnings and errors).

	Line number	Description	Compiler Messages	
			Line number	Description
1.				
2.				
..				

2. Debug the code and make sure that both the main code and the test code run properly (as far as it was completed).
3. Implement a function to calculate the standard deviation of the set of grades:

$$s = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2},$$

where N is the size of the sample, x_i the value of item i and \bar{x} the average of the sample.

4. Implement a test for this function in the file `test.c`.

5. Complete the routine `fill_histogram` to calculate the histogram. The histogram has 12 intervals:

1 : the values 0.0-0.5
2 : the values 0.5-1.5
.....
11 : the values 9.5-9.9
12 : the value 10.0

Implement a new function `print_histogram` to print the histogram to the screen in the following format:

```
0.0-0.5 : 12 #####  
0.5-1.5 : 34 #####  
1.5-2.5 : 33 #####  
2.5-3.5 : 50 #####  
...  
etc.,
```

where first the interval range is printed, then a semi-colon, followed by the number of items in the interval and a 'bar' composed of pound keys to visualise the number of items in the interval. Scale the length of this bar in such a way that the longest bar is 40 characters long.

6. Design and implement a test for this function.

Useful information:

Pelles C compiler: <http://www.smorgasbordet.com/pellec/>