12 points (B)

……………………………………..
Name and surname

During the test you can only use cplusplus.com and cppreference.com, jug.dpieczynski.pl and your own homeworks only from E-kursy where it had been uploaded during the exam. Any other source, even your own code from file or email or … is not allowed.

Task 0 (2 points)

Split your project into multiple files. Your project should contain at least two source files (the one containing main function and one other) and one header file. Avoid copying the data when not required (use reference or const reference instead).

Task 1 (3 points)

Read the provided .txt file into appropriate data structures knowing that every line contains 8 columns that are separated with comma (**,** characters).
The first column contains students' names, second column students' surnames and the seven remaining columns contain grades scored for seven different tests.
Split name and surname. Use container to store grades.

Task 2 (3 points)

Knowing that the total number of grades that a student can get is 70, write a function that will calculate a final grade for the given student. Use appropriate standard library function to sum (accumulate) the points.

`Std::accumulate` (check cplusplus.com or cppreference.com)

Use the data from the table below to determine the final grade. Print the students names and surnames along with the result of a calculation.

Percent of total points	Grade
91 - 100	5.0
81 - 90	4.50
71 - 80	4.25
61 - 70	3.50
51 - 60	3.25
0 - 50	2.25

Example call:

```
// Grab every student from some container
for (auto &student : students_container) {
   // Calculate grade for every student
   double grade = calculate_grade(student);
   // Print the name with the grade
}
```

Example output:

```
John Adams 4.5
...
```

Task 3 (4 points)

Extend your data structure so that every student can have a final grade assigned. Calculate and assign the final grade for every student using the function from previous task.

Create a function that will takes student container and a boolean parameter: Boolean parameter called “*by_grade*” If “*by_grade*” is true the function should sort students by final grade. If “*by_grade*” is false the function should sort the students by surname. Use appropriate function from standard library for the sorting.

Example call:

```
sort_students_by_grade(students_container, true);
//print name and surname of sorted students
```