First semester complex project – C#

Participants of the project: **Pallaga Pál**, **Szabó Hubertusz**

First, I created a console app in the C# programing language, I started with making a class object with the name “Sportolo”, this class has all necessary attributes from the file which are the following (Note that all variables are Hungarian in the code):

1. Placement (helyezes)
2. Result (eredmeny, “the result of their throw”)
3. Name
4. Country Code (orszagkod, “3 letter code e.g.: JAP for Japan”)
5. Place (helyszin, “where the throw took place”)
6. Year (ev)
7. Month (honap)
8. Day (nap, “the last three are the date of the throw”)

After that I made a method that tells the user that they must press enter to continue, which may exit the program if it is the last part, the method is called Var() (as in wait in hungarian).

Now I made a sportolok named list which will contain all sportsmen with their attributes and another list for the results called eredmenyek, this will be used for some complicated programming shenanigans later.

Then I had to read the file into a variable called “f”, which was a basic text file, where all attributes are separated by semicolons (';'). After which I split up the text by the characters ‘;’, ‘.’ and ‘\n’ -> this is an enter character and put that into another variable called “g”.

I created a for cycle which starts at i = 6, while i is smaller than “g”, then 8 is added to i (due to a record being 8 attributes).

In this cycle I did the following:

1. (i+1)th element of the g variable will have all colons replaced by dots (‘,’ -> ‘.’, this is because that will be a double type variable later)
2. I make an instance of the class Sportolo, called sportolo (notice that the second is a variable because it is a lowercase letter) and I assigned each attribute to i plus whichever number we need from the record e.g.: i+1 for the results, whilst I convert the g variable strings to the needed type, such as double for i+1.
3. I add the sportolo variable to the list sportolok
4. I then add the sportolo.eredmeny to the eredmenyek list.

I then write out into the console 4 mathematical statistics from the two lists, such as the average of the results, with it being rounded to 2 decimal places (eredmenyek.Average(), 2)

The 2nd one is the smallest result (eredmenyek.Min()) with the person who had that result, which I calculated with the following (sportolok[eredmenyek.IndexOf(eredmenyek.Min())].nev)

The 3rd is the biggest result which is nearly the same as the last one (eredmenyek.Max()) and (sportolok[eredmenyek.IndexOf(eredmenyek.Max())].nev)

The last one is the median of the results (if (eredmenyek.Count() % 2 == 0) median = (eredmenyek[eredmenyek.Count() / 2] + eredmenyek[eredmenyek.Count() / 2 - 1]) / 2;

else median = eredmenyek[eredmenyek.Count() / 2]; ) then I write the median out into the console

I use the Var() method to signal the next part of the code

The last part of the code is a seemingly endless loop which will end if you answer one of the requests with nothing, which the program tells you. This cycle is for searching for most of the attributes in the sportsmen, which when the program finds any matches then it will write the sportsman and all their attributes

The last part is to wait again where when the user presses enter, the program ends