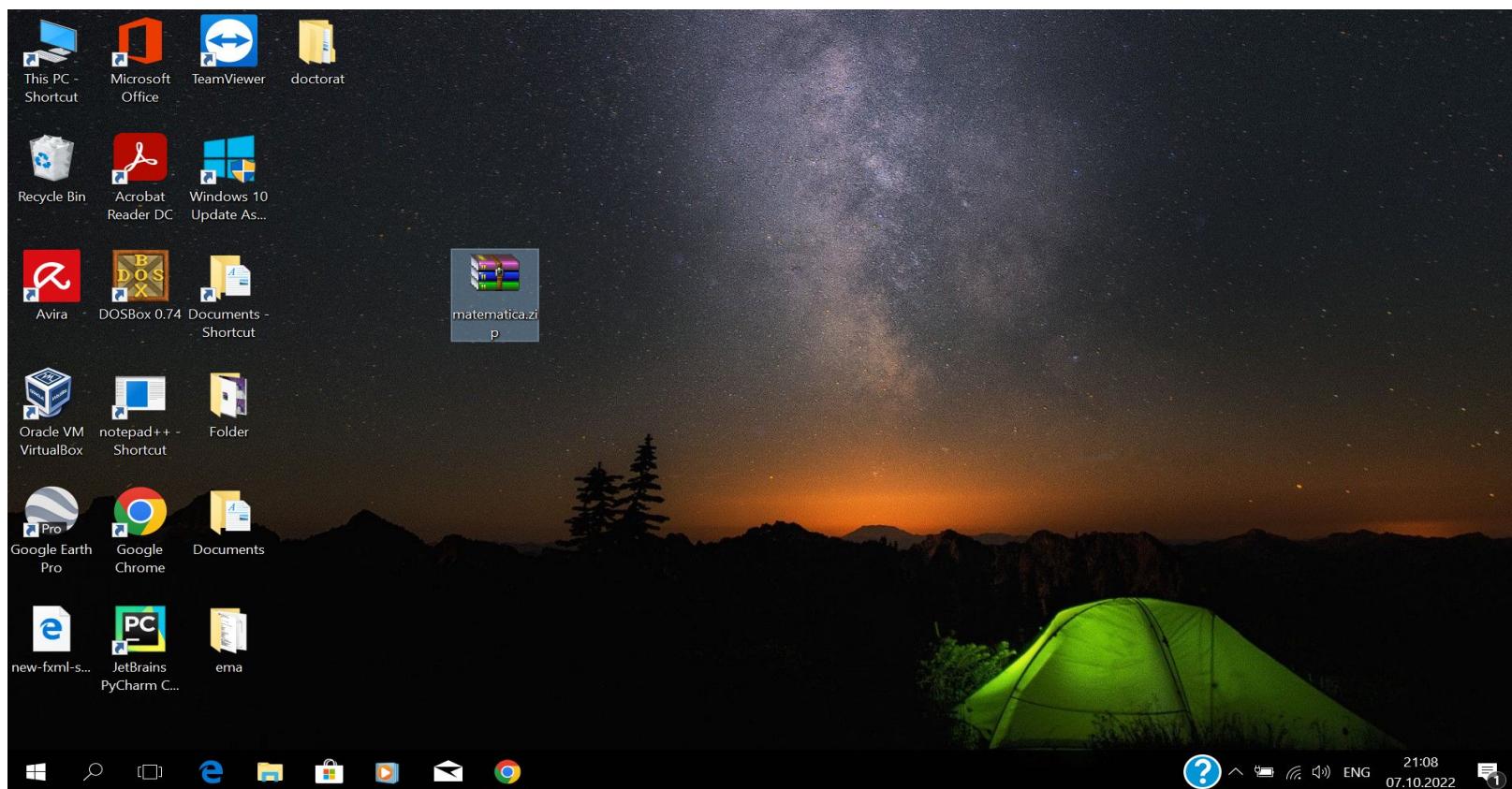
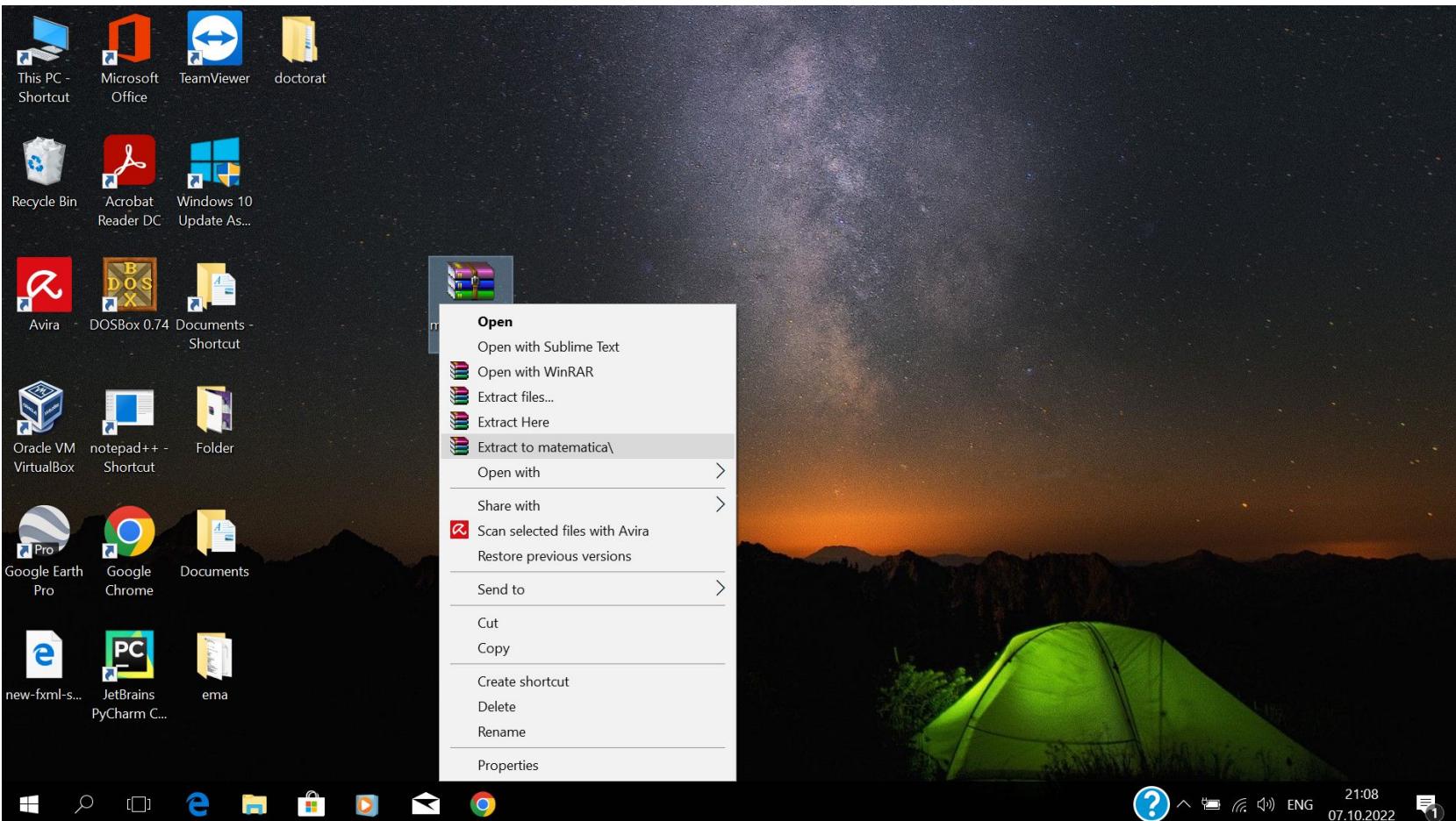


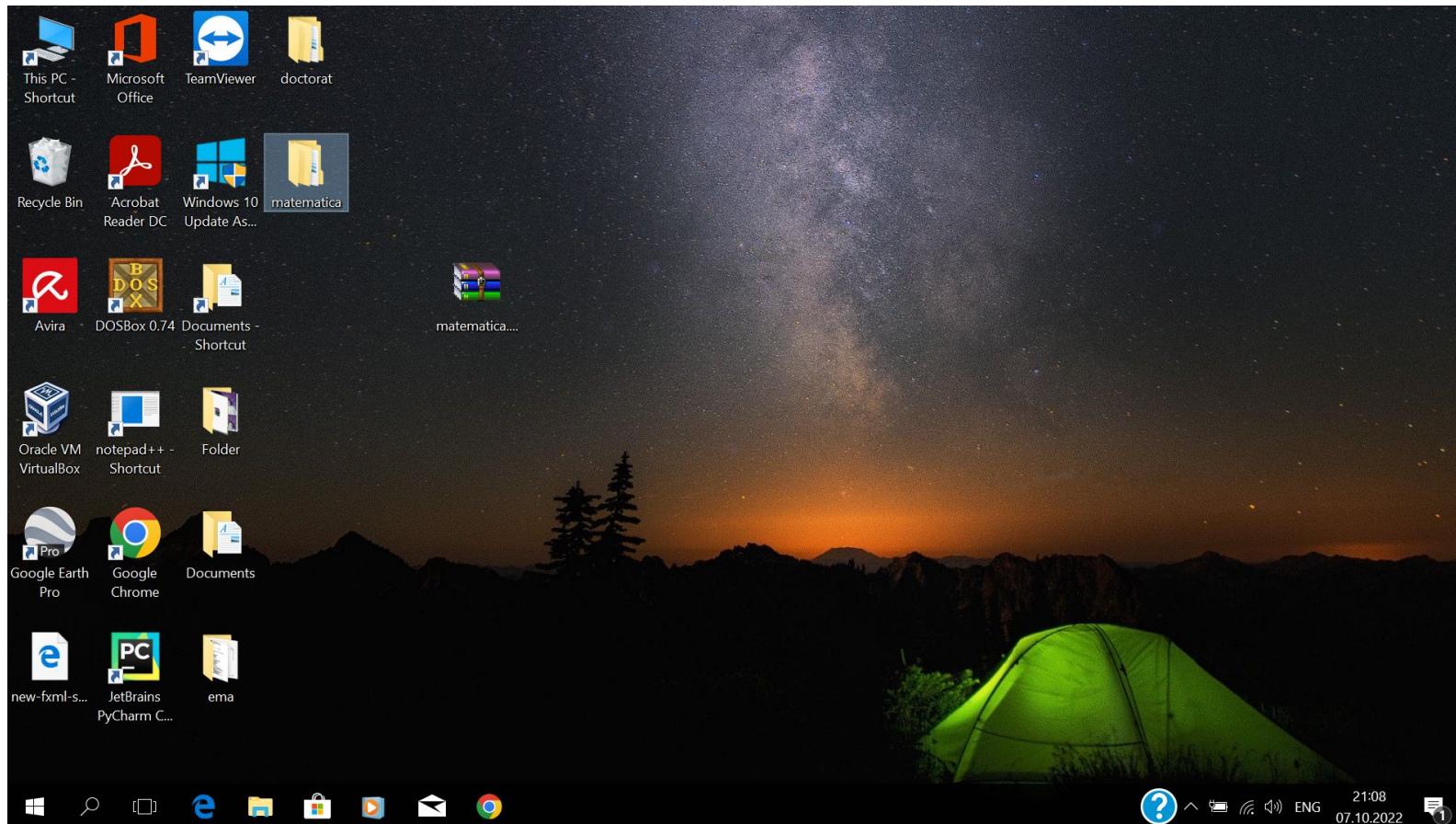
Veti primi o arhiva .zip cu programul pe care l-am scris la seminar.



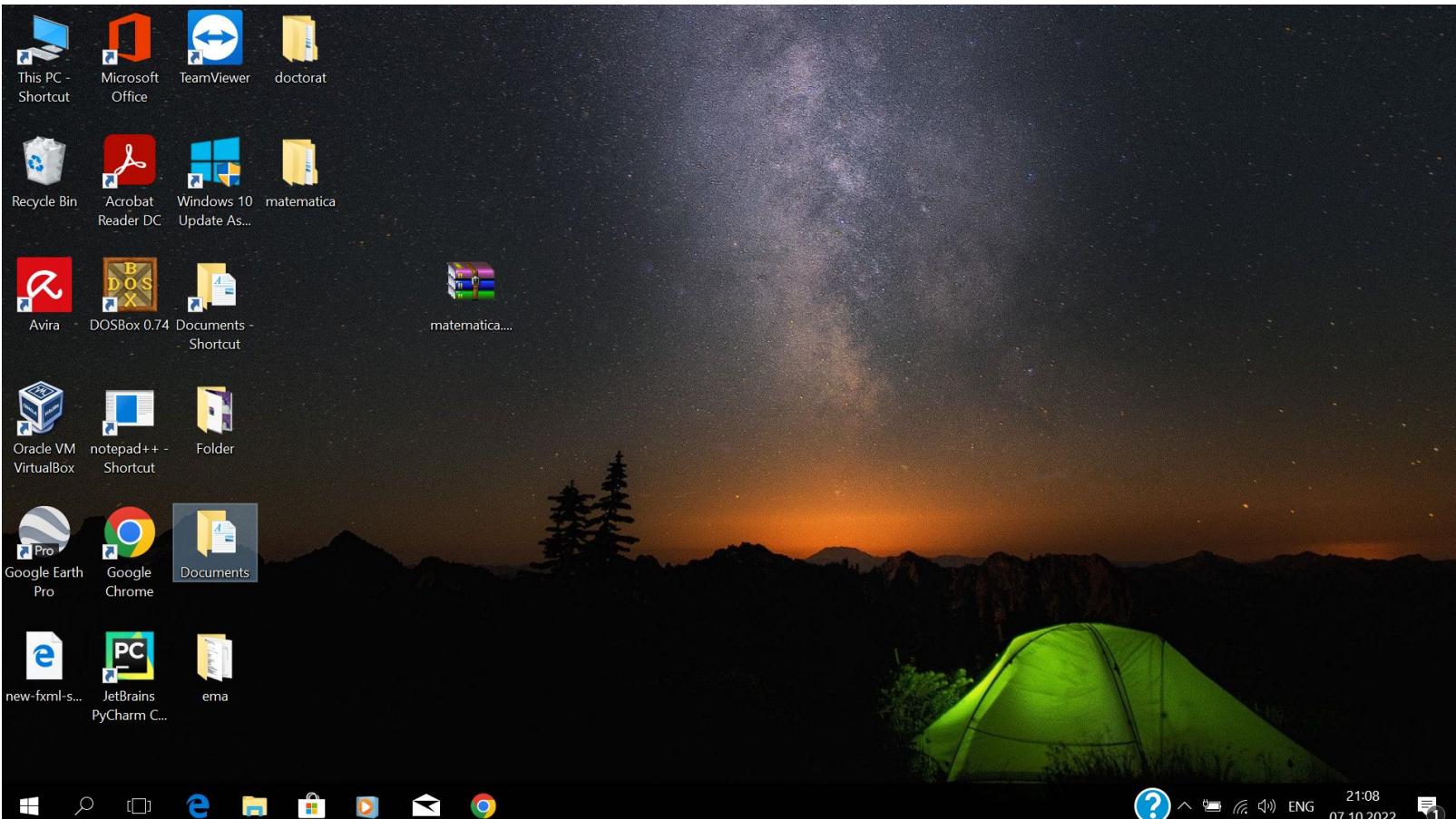
Dati click dreapta pe arhiva si selectati "Extract to matematica\"



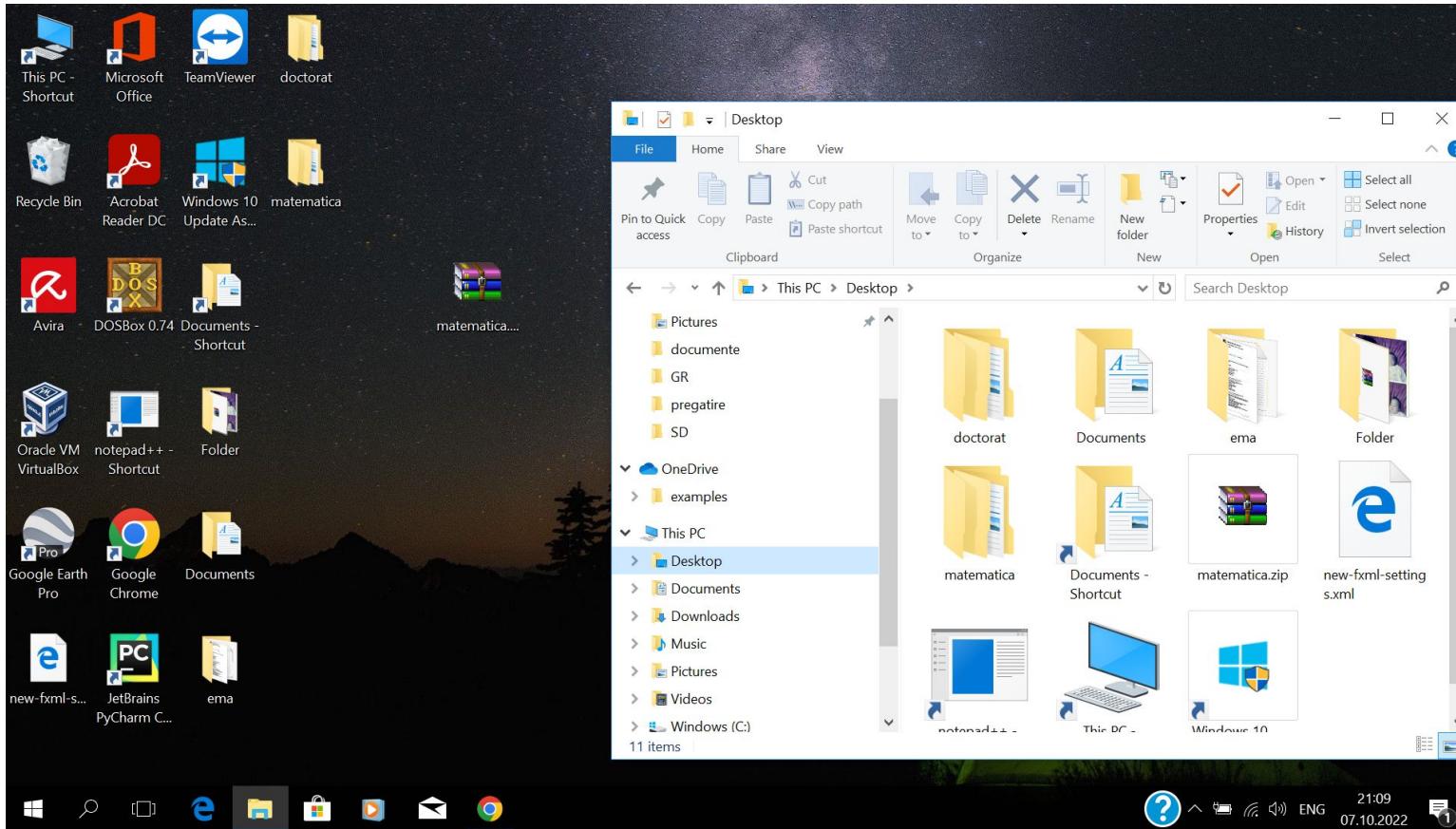
Veti vedea ca va aparea un fisier numit matematica.



Mergeti apoi in fisierul Documente si deschideti-l.

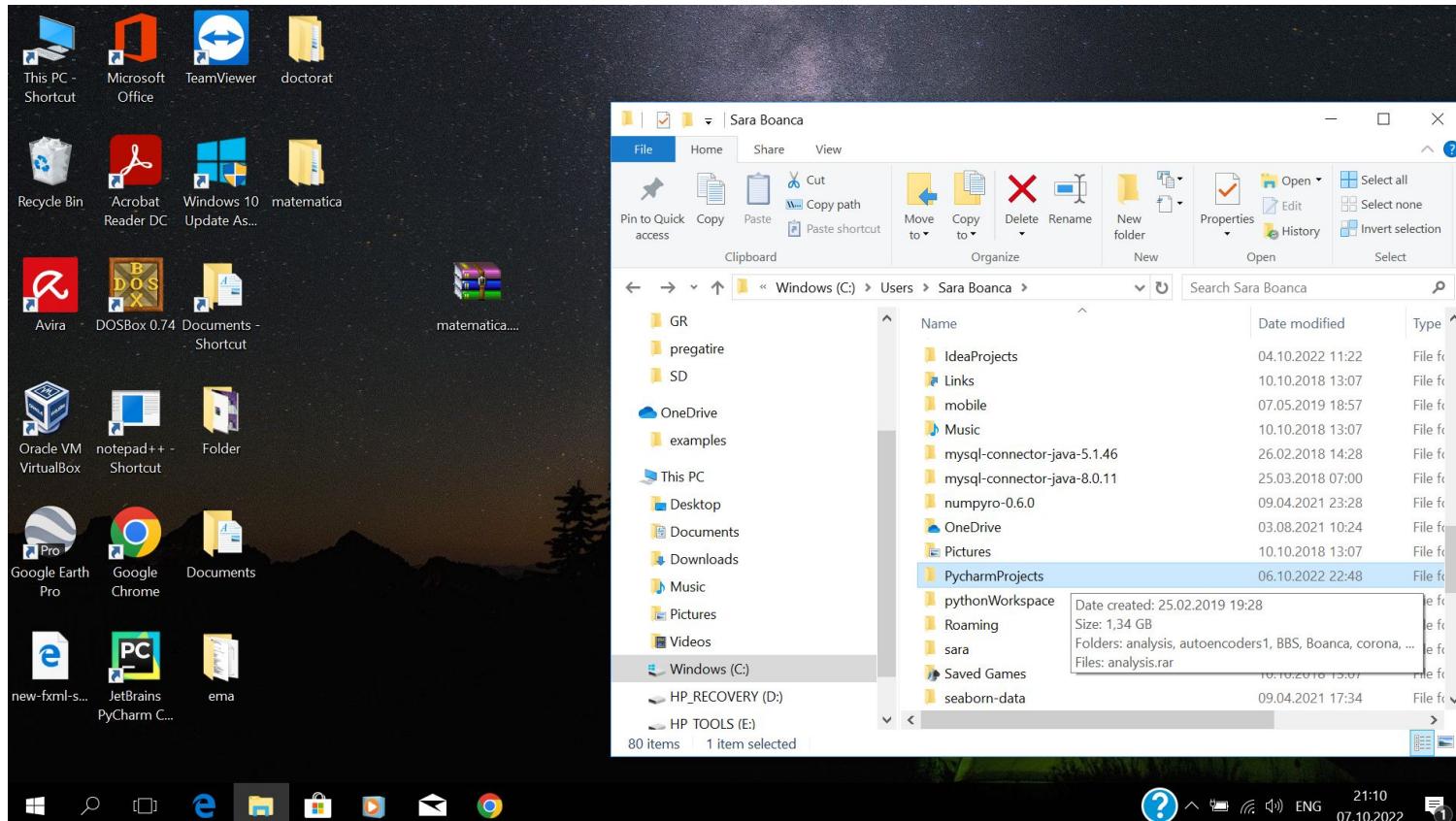


Acum va trebui sa gasim locul unde sunt proiectele noastre PyCharm ca sa adaugam fisierul matematica acolo.

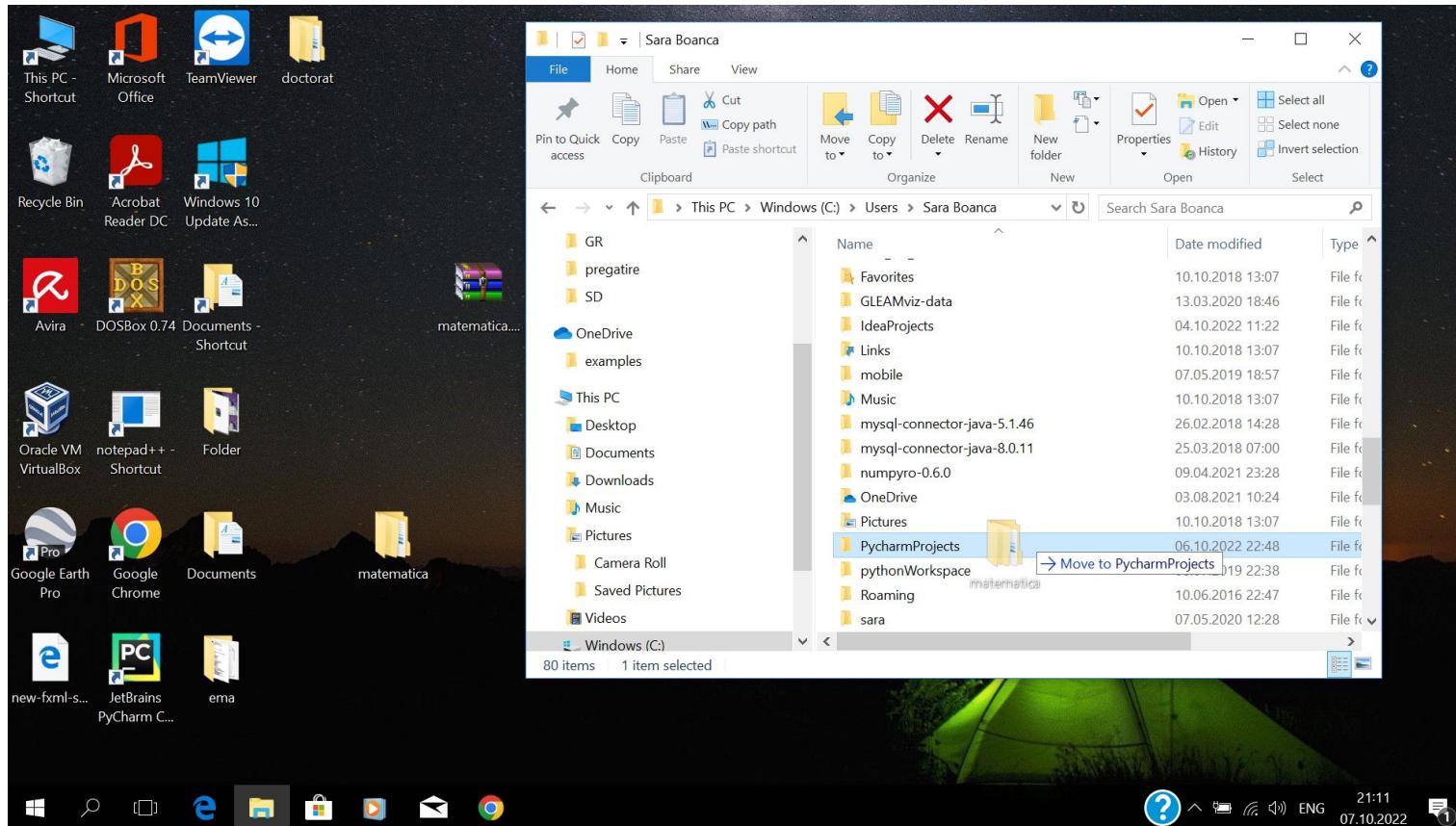


La mine este in Windows C -> Users -> SaraBoanca -> PycharmProjects.

Se poate ca la voi calea spre locatia fisierului PycharmProjects sa fie diferita, dar in principiu cam pe aici ar trebui sa fie. Va trebui sa cautati exact unde e fisierul PycharmProjects la voi pe calculator.

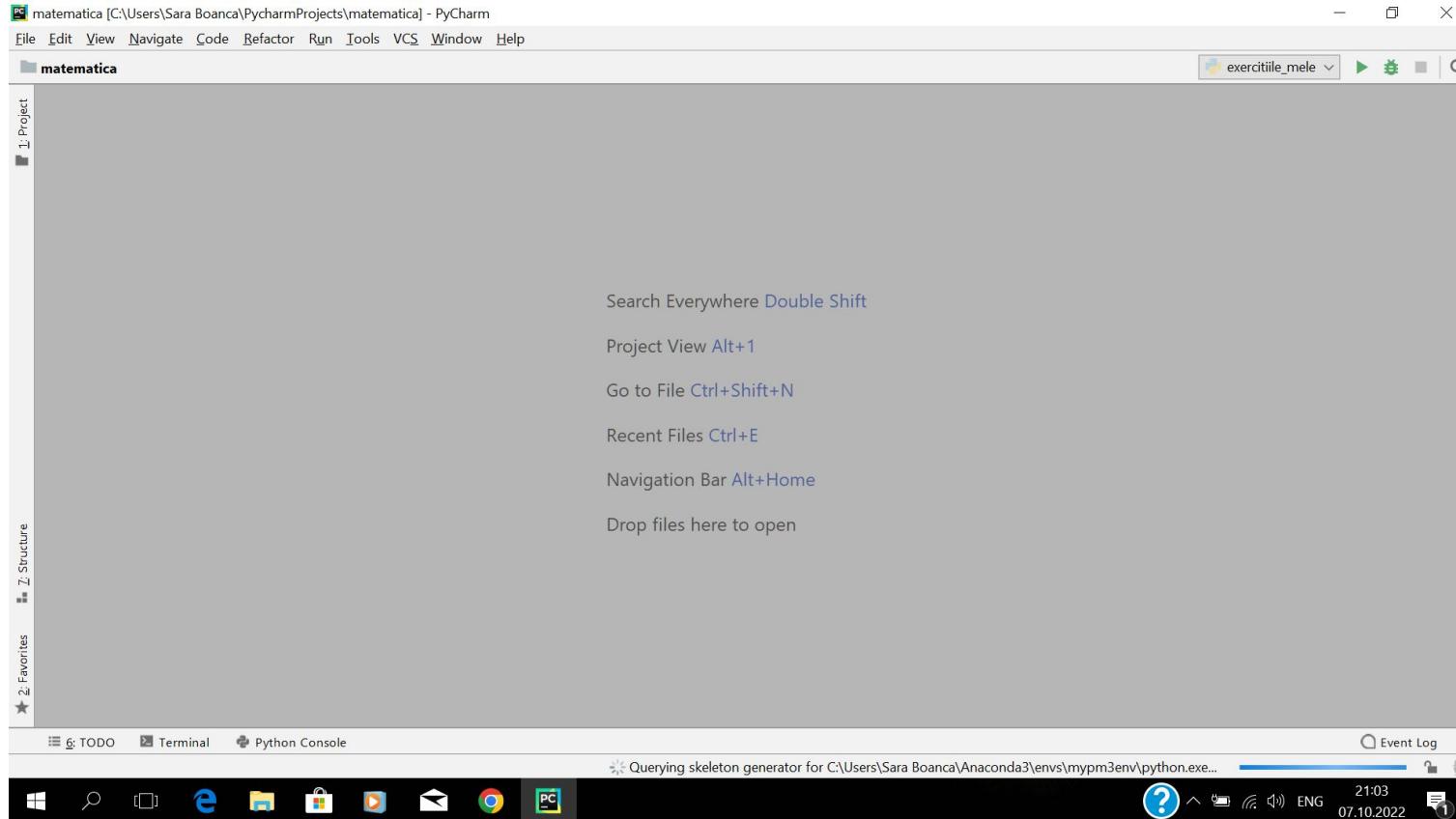


Cand ati gasit locatia fisierului PycharmProjects, copiati fisierul matematica in fisierul PycharmProjects.  
Dupa ce ati facut asta, puteti deschide PyCharm-ul.

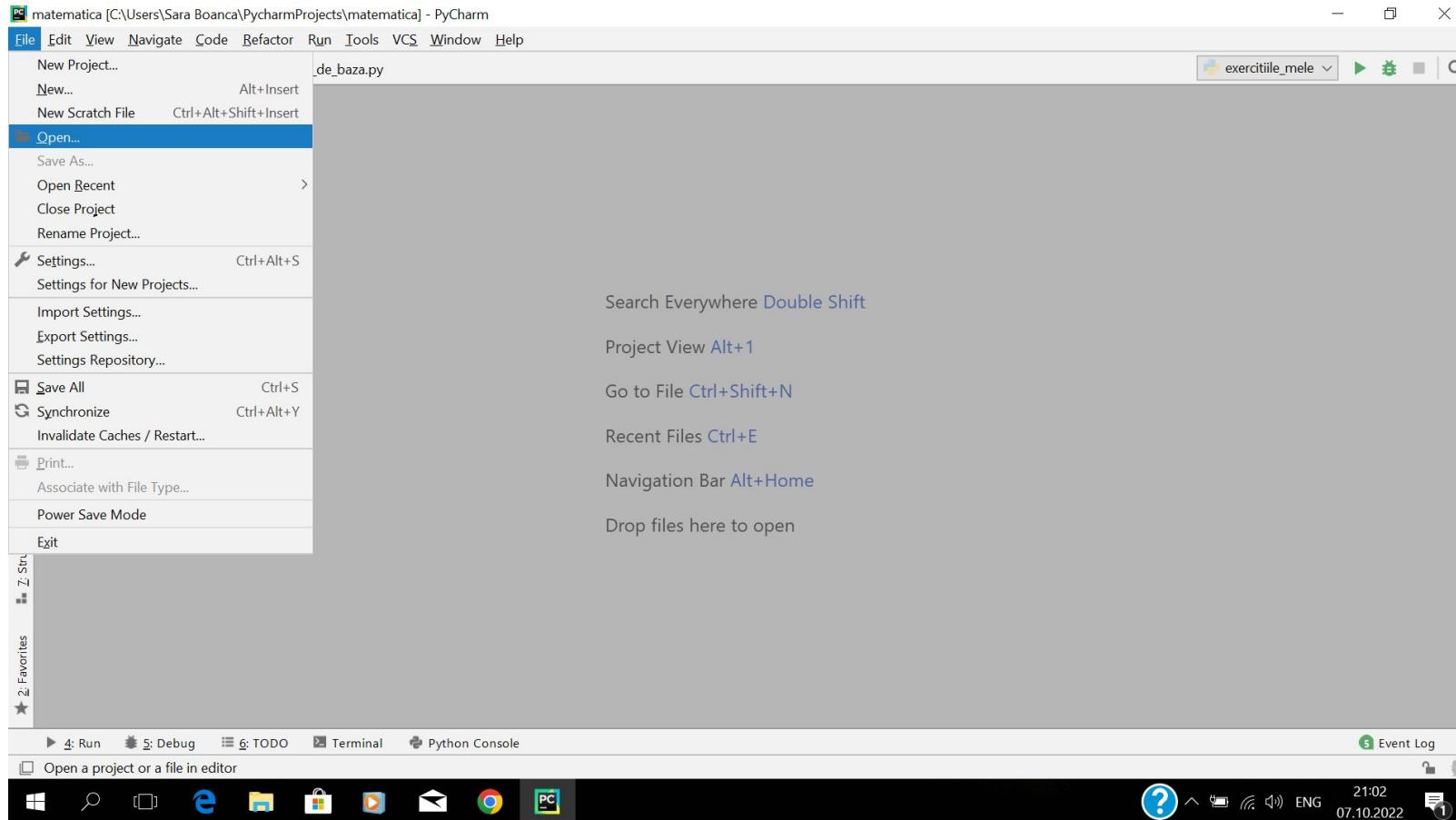


Veti vedea ca probabil la inceput nu apare nimic.

Nu e o problema. Acum vom deschide fisierul matematica cu programul scris la seminar.



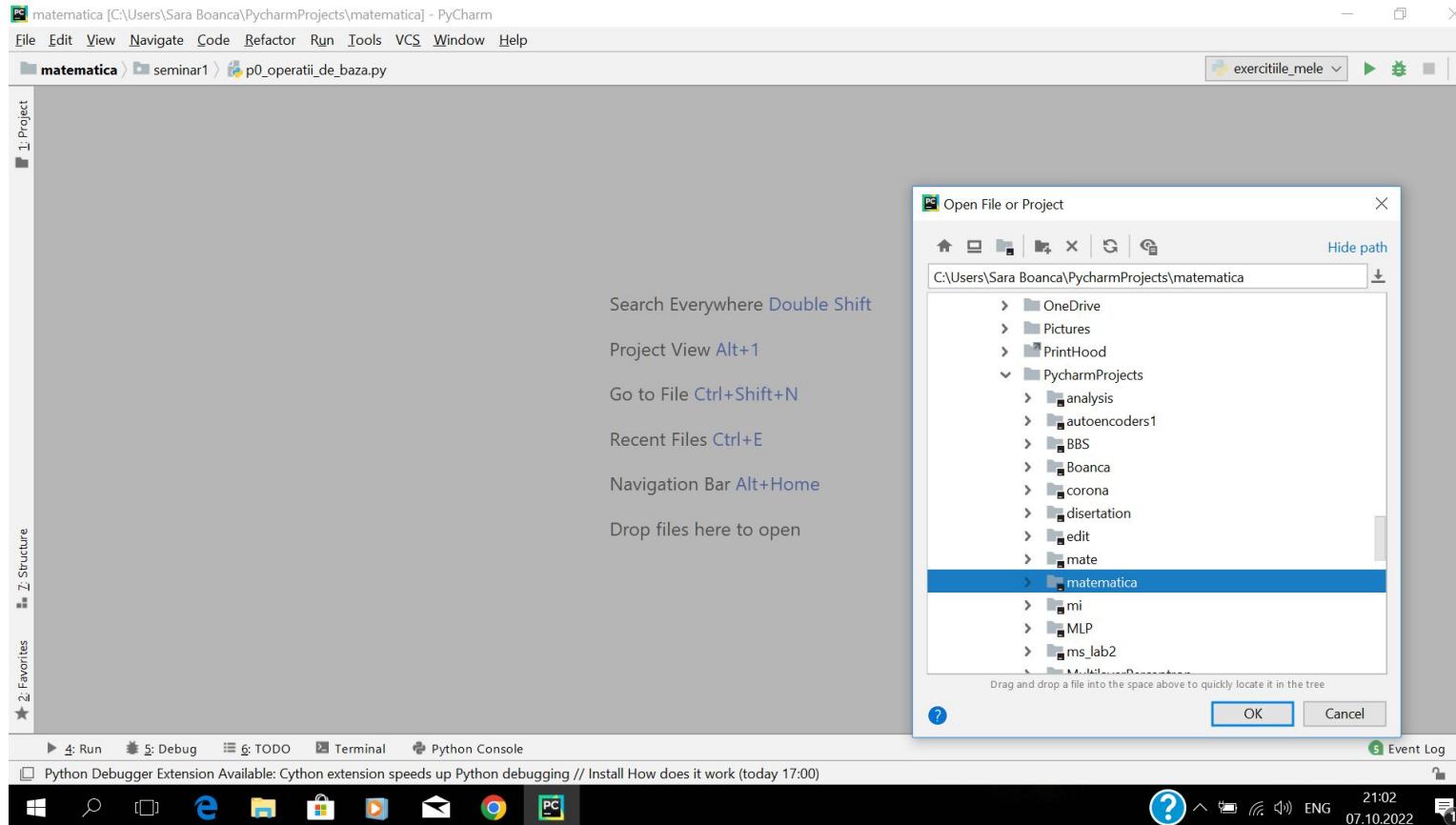
# Mergeti pe File si selectati optiunea Open.



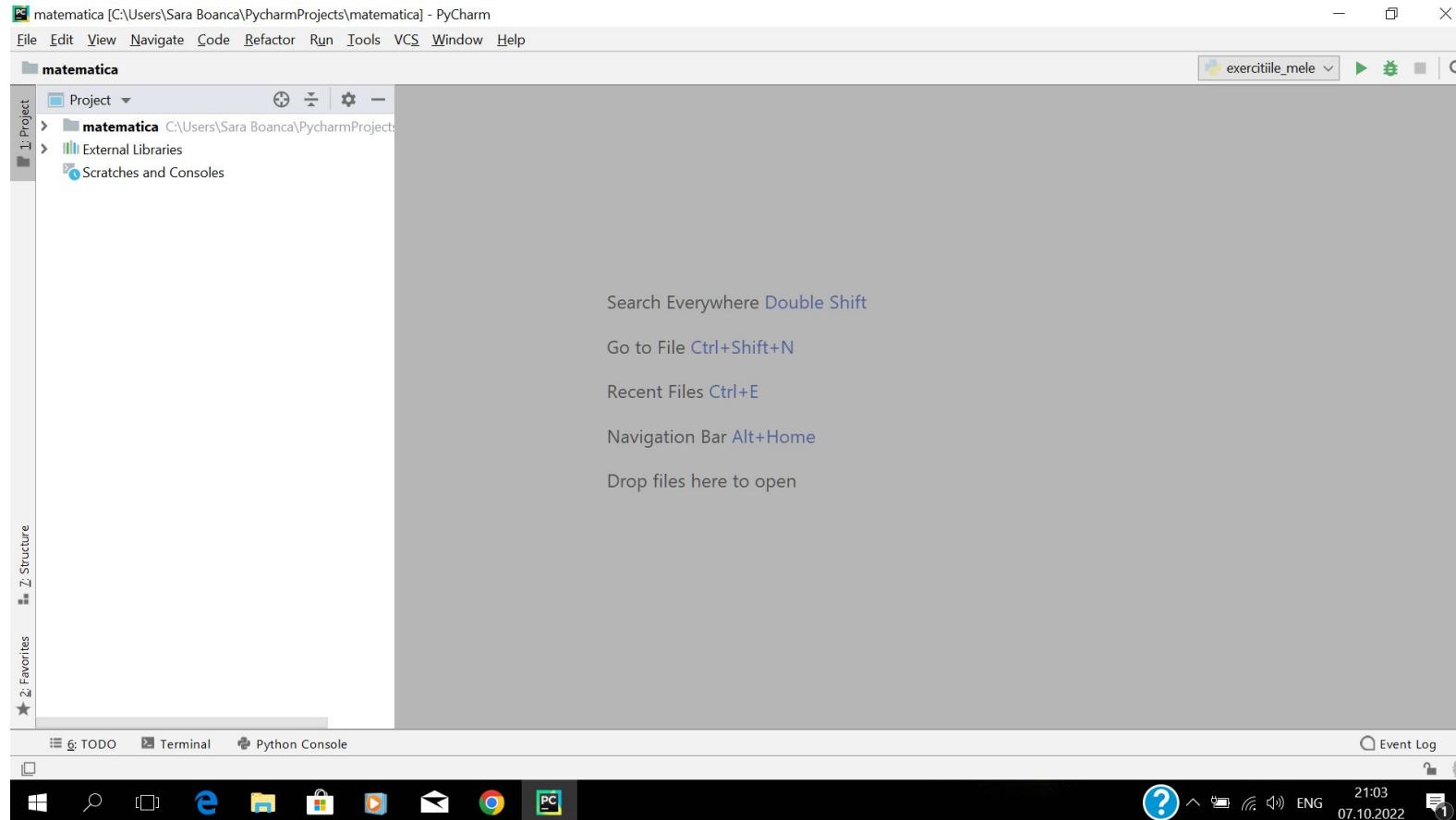
Veti vedea ca se deschide un selector de fisiere.

Voi trebuie sa mergeți la PycharmProjects si de acolo sa selectati fisierul matematica.

Dati ok.

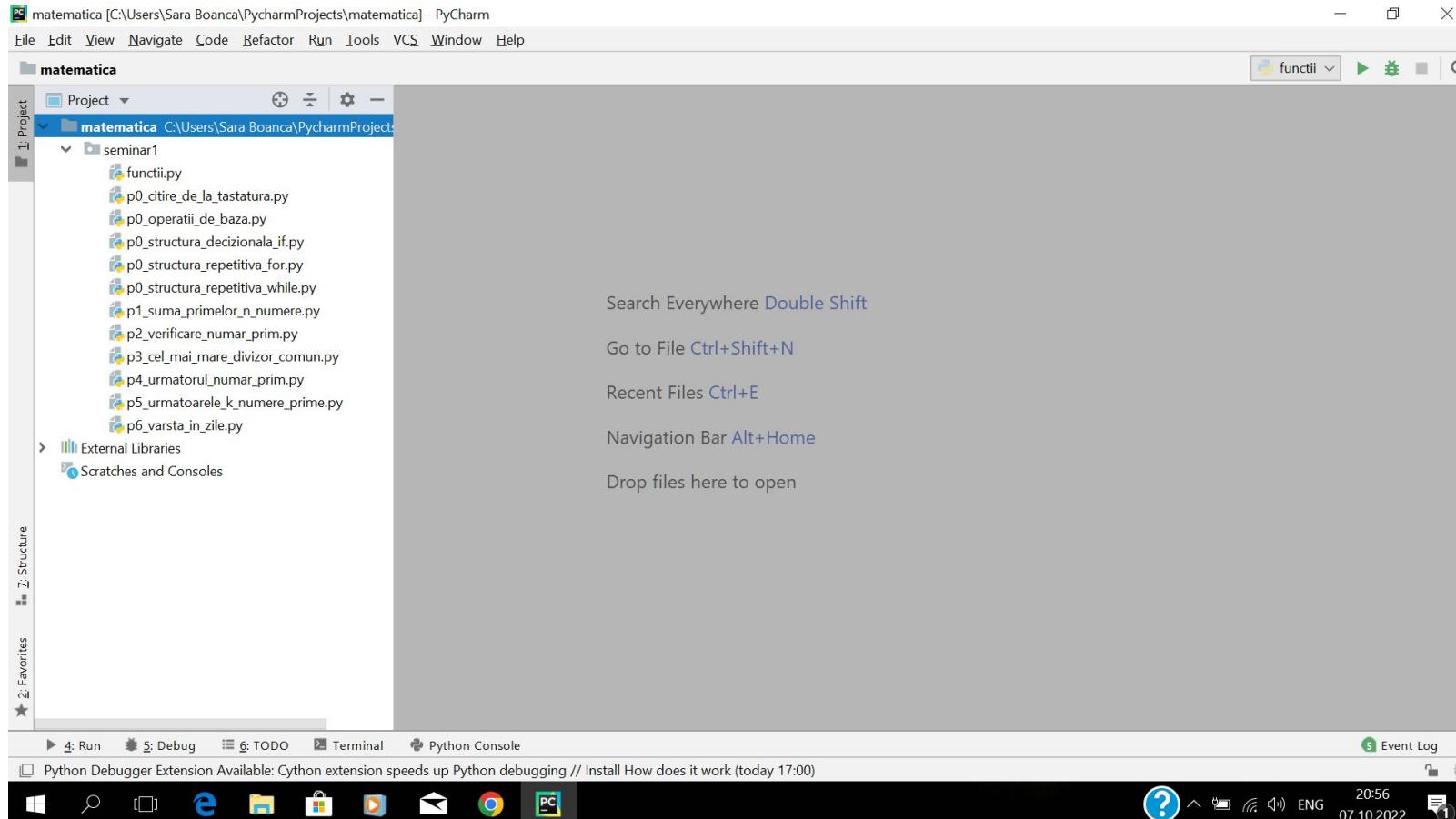


Acum am deschis fisierul matematica in PyCharm. Ar trebui sa vedeti ceva de genul asta.



Daca dati click pe matematica, veti vedea ca apare fisierul seminar1.

Daca dati click pe el, vor aparea fisierele cu cod, unde am rezolvat problemele de la seminar.



Dati dublu click pe fisierul pe care vreti sa il deschideti.

The screenshot shows the PyCharm IDE interface. The title bar indicates the project is named "matematica" and the file being edited is "p0\_citire\_de\_la\_tastatura.py". The menu bar includes File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, and Help. The toolbar has icons for Run, Debug, TODO, Terminal, and Python Console. The Project tool window on the left shows a tree structure with the "matematica" project containing a "seminar1" folder with various Python files, including "p0\_citire\_de\_la\_tastatura.py" which is currently selected and highlighted in blue. The main editor window displays the following code:

```
#citim de la tastatura un text
sir_de_caractere = input("Scripteti un text aici si apasati Enter: ")
print("Textul introdus este", sir_de_caractere)

#citim de la tastatura un numar
numar = int(input("Acum introduceti un numar si din nou apasati Enter: "))
print("Numarul introdus este", numar)
```

The status bar at the bottom shows the Python Debugger Extension Available message, the current time (7:38), encoding (CRLF), file encoding (UTF-8), indentation settings (4 spaces), and the date (07.10.2022). The system tray icons include a question mark, battery, signal strength, volume, language (ENG), and a notification icon.

## Puteti deschide fisierele, sa vedeti ce e in ele.

The screenshot shows the PyCharm IDE interface. The title bar indicates the current file is `p0_operatii_de_baza.py`. The left sidebar displays the project structure under the `matematica` root, including files such as `functii.py`, `p0_citire_de_la_tastatura.py`, and `p0_operatii_de_baza.py`. The main editor window shows the code for `p0_operatii_de_baza.py`, which includes comments explaining the use of triple quotes for multi-line comments and simple vs double quotes for single-line comments. It also demonstrates printing to the console and performing arithmetic operations on variables `a` and `b`.

```
#Acesta este un comentariu. Observati ca incepe cu #. Comentariile NU se executa cand rulam programul.
"""
Daca vrem sa comentam mai multe lini, le punem intre ghilimele triple.
"""

"""
Ghilimele triple folosite pentru comentariu pot fi simple('') sau duble("")
In comentariul anterior am folosit ghilimele duble.
In comentariul acesta am folosit ghilimele simple.

"""
#Asa tiparim un mesaj in consola
print("Hello!")

#Asa asignam valori unor variabile
a = 10
b = 2

#Facem operatii cu variabilele noastre
#Observati ca putem da ce nume vrem noi variabilelor si este important sa fie sugestive
adunare = a+b
print("a + b =", adunare)

scadere = a-b
print("a - b =", scadere)

inmultire = a*b
print("a * b =", inmultire)
```

At the bottom, the status bar shows the Python Debugger Extension is available, and the system tray indicates it's 20:57 on 07.10.2022.

Daca vreti sa rulati codul dintr-un fisier, mergeti in fereastra unde e scris codul si dati click dreapta acolo. Veti vedea ca se deschide un meniu. Selectati Run.

The screenshot shows the PyCharm IDE interface. The main window displays a Python file named `p0_operatii_de_baza.py`. The code in the file is as follows:

```
#Acesta este un comentariu. Observati ca incepe cu #. Comentariile NU se executa cand ruleam programul.  
'''  
Daca vrem sa comentam mai multe linii,  
le punem intre ghilimele triple.  
'''  
Ghilimele triple folosite pentru comentariu pot  
sa contin multe linii de comentariu.  
In comentariul anterior am folosit ghilimele duble.  
In comentariul acesta am folosit ghilimele simple.  
'''  
  
#Asa tiparim un mesaj in consola  
print("Hello!")  
  
#Asa asignam valori unor variabile  
a = 10  
b = 2  
  
#Facem operatii cu variabilele noastre  
#Observati ca putem da ce nume vrem noi variabilei  
adunare = a+b  
print("a + b =", adunare)  
  
scadere = a-b  
print("a - b =", scadere)  
  
inmultire = a*b  
print("a * b =", inmultire)
```

A context menu is open over the code, specifically over the first few lines. The menu items include:

- Copy Reference
- Paste
- Paste from History...
- Paste without Formatting
- Column Selection Mode
- Find Usages
- Refactor
- Folding
- Go To
- Generate...
- Run 'p0\_operatii\_de\_baza'
- Debug 'p0\_operatii\_de\_baza'
- Create 'p0\_operatii\_de\_baza...'
- Show in Explorer
- Open in Terminal
- Local History
- Execute Line in Console
- Run File in Console
- Compare with Clipboard
- File Encoding
- Create Gist...

The "Run 'p0\_operatii\_de\_baza'" option is highlighted with a yellow background, indicating it is the selected action. The PyCharm interface also includes a Project tool window on the left, a Python Console at the bottom, and various status indicators at the bottom right.

Observati cum codul din fisierul la care i-am dat Run se executa in consola.

The screenshot shows the PyCharm IDE interface with the following details:

- Project:** matematica
- File:** C:\Users\Sara Boanca\PycharmProjects\matematica - ...\\seminar1\\p0\_operatii\_de\_baza.py [matematica] - PyCharm
- Code Editor:** The file p0\_operatii\_de\_baza.py contains the following code:

```
#Acesta este un comentariu. Observati ca incepe cu #. Comentariile NU se execute cand rulam programul.
"""
Daca vrem sa comentam mai multe linii,
le punem intre ghilimele triple.
"""

"""
Ghilimele triple folosite pentru comentariu pot fi simple('') sau duble("")
In comentariul anterior am folosit ghilimele duble.
In comentariul acesta am folosit ghilimele simple.

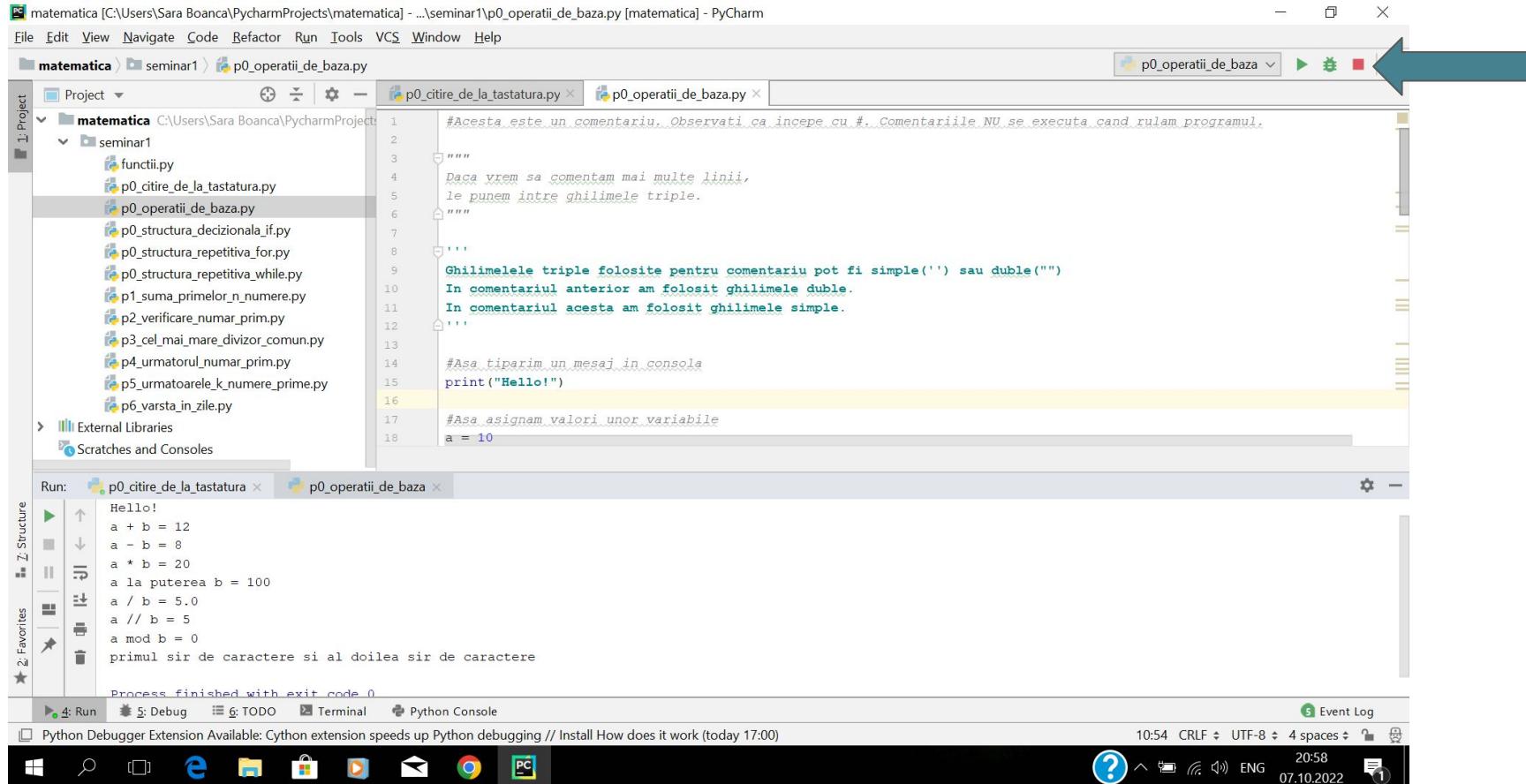
"""
#Asa tiparim un mesaj in consola
print("Hello!")

#Asa asignam valori unor variabile
a = 10
```
- Run Tab:** Shows the output of running the script:

```
Hello!
a + b = 12
a - b = 8
a * b = 20
a la puterea b = 100
a / b = 5.0
a // b = 5
a mod b = 0
primul sir de caractere si al doilea sir de caractere
```

Process finished with exit code 0
- Bottom Status Bar:** Includes icons for Run, Debug, TODO, Terminal, Python Console, Event Log, and system status (10:54, CRLF, UTF-8, 4 spaces, ENG, 20:58, 07.10.2022).

Daca vreti sa opriti manual executia programului, dati click pe patratul rosu din partea dreapta.



matematica [C:\Users\Sara Boanca\PycharmProjects\matematica] - ...\\seminar1\\p0\_operatii\_de\_baza.py [matematica] - PyCharm

File Edit View Navigate Code Refactor Run Tools VCS Window Help

matematica > seminar1 > p0\_operatii\_de\_baza.py

Project

matematica C:\Users\Sara Boanca\PycharmProjects\matematica  
└── seminar1  
 ├── functii.py  
 ├── p0\_citire\_de\_la\_tastatura.py  
 ├── p0\_operatii\_de\_baza.py  
 ├── p0\_structura\_decizionala\_if.py  
 ├── p0\_structura\_repetitiva\_for.py  
 ├── p0\_structura\_repetitiva\_while.py  
 ├── p1\_suma\_primerelor\_n\_numere.py  
 ├── p2\_verificare\_numar\_prim.py  
 ├── p3\_cel\_mai\_mare\_divizor\_comun.py  
 ├── p4\_urmaritorul\_numar\_prim.py  
 ├── p5\_urmatoarele\_k\_numere\_prime.py  
 └── p6\_varsta\_in\_zile.py

External Libraries

Scratches and Consoles

Run: p0\_citire\_de\_la\_tastatura x p0\_operatii\_de\_baza x

```
#Acesta este un comentariu. Observati ca incepe cu #. Comentariile NU se execute cand rulam programul.  
'''  
Daca vrem sa comentam mai multe linii,  
le punem intre ghilimele triple.  
'''  
Ghilimele triple folosite pentru comentariu pot fi simple('') sau duble("")  
In comentariul anterior am folosit ghilimele duble.  
In comentariul acesta am folosit ghilimele simple.  
...  
#Asa tiparim un mesaj in consola  
print("Hello!")  
#Asa asignam valori unor variabile  
a = 10
```

Hello!  
a + b = 12  
a - b = 8  
a \* b = 20  
a la puterea b = 100  
a / b = 5.0  
a // b = 5  
a mod b = 0  
primul sir de caractere si al doilea sir de caractere

Process finished with exit code 0

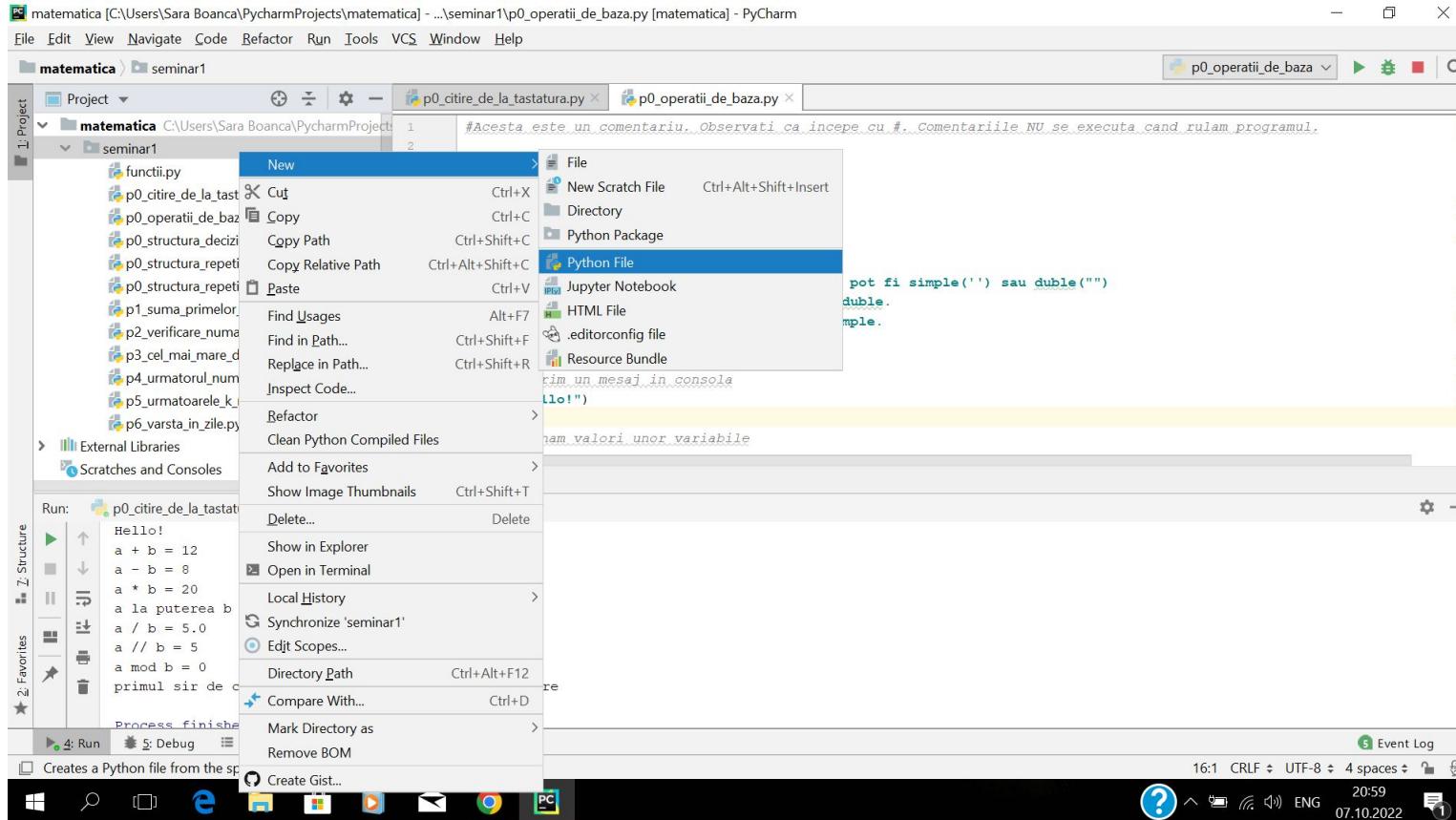
Run Debug TODO Terminal Python Console

Python Debugger Extension Available: Cython extension speeds up Python debugging // Install How does it work (today 17:00)

10:54 CRLF UTF-8 4 spaces Event Log

20:58 07.10.2022

Daca vreti sa creati un nou fisier gol, mergeti in meniul din stanga pe seminar1.  
Dati click dreapta pe el si selectati New -> Python File.



Vi se va cere sa denumiti noul fisier python creat. Eu l-am denumit exercitiile\_mele.  
Dupa ce scrieti numele fisierului dati Ok sau Enter.

matematica [C:\Users\Sara Boanca\PycharmProjects\matematica] - ...\\seminar1\\p0\_operatii\_de\_baza.py [matematica] - PyCharm

File Edit View Navigate Code Refactor Run Tools VCS Window Help

matematica seminar1

Project

matematica C:\Users\Sara Boanca\PycharmProjects\matematica

seminar1

- functii.py
- p0\_citire\_de\_la\_tastatura.py
- p0\_operatii\_de\_baza.py
- p0\_structura\_decizionala\_if.py
- p0\_structura\_repetitiva\_for.py
- p0\_structura\_repetitiva\_while.py
- p0\_suma\_primerelor\_n\_numere.py
- p2\_verificare\_numar\_prim.py
- p3\_cel\_mai\_mare\_divizor\_comun.py
- p4\_urmaritorul\_numar\_prim.py
- p5\_urmatoarele\_k\_numere\_prime.py
- p6\_varsta\_in\_zile.py

External Libraries

Scratches and Consoles

New Python file

Name: exercitiile\_mele

Kind: Python file

OK Cancel

#Acesta este un comentariu. Observati ca incepe cu #.. Comentariile NU se executa cand rulam programul.

"""

Daca vrem sa comentam mai multe linii, le punem intre ghilimele triple.

"""

'''

Ghilimele triple folosite pentru comentariu pot fi simple('') sau duble("")

In comentariul anterior am folosit ghilimele duble.

In comentariul anterior am folosit ghilimele simple.

#Asa tipa

print ("Hello")

#Asa asemanator

a = 10

Hello!

a + b = 12

a - b = 8

a \* b = 20

a la puterea b = 100

a / b = 5.0

a // b = 5

a mod b = 0

primul sir de caractere si al doilea sir de caractere

Process finished with exit code 0

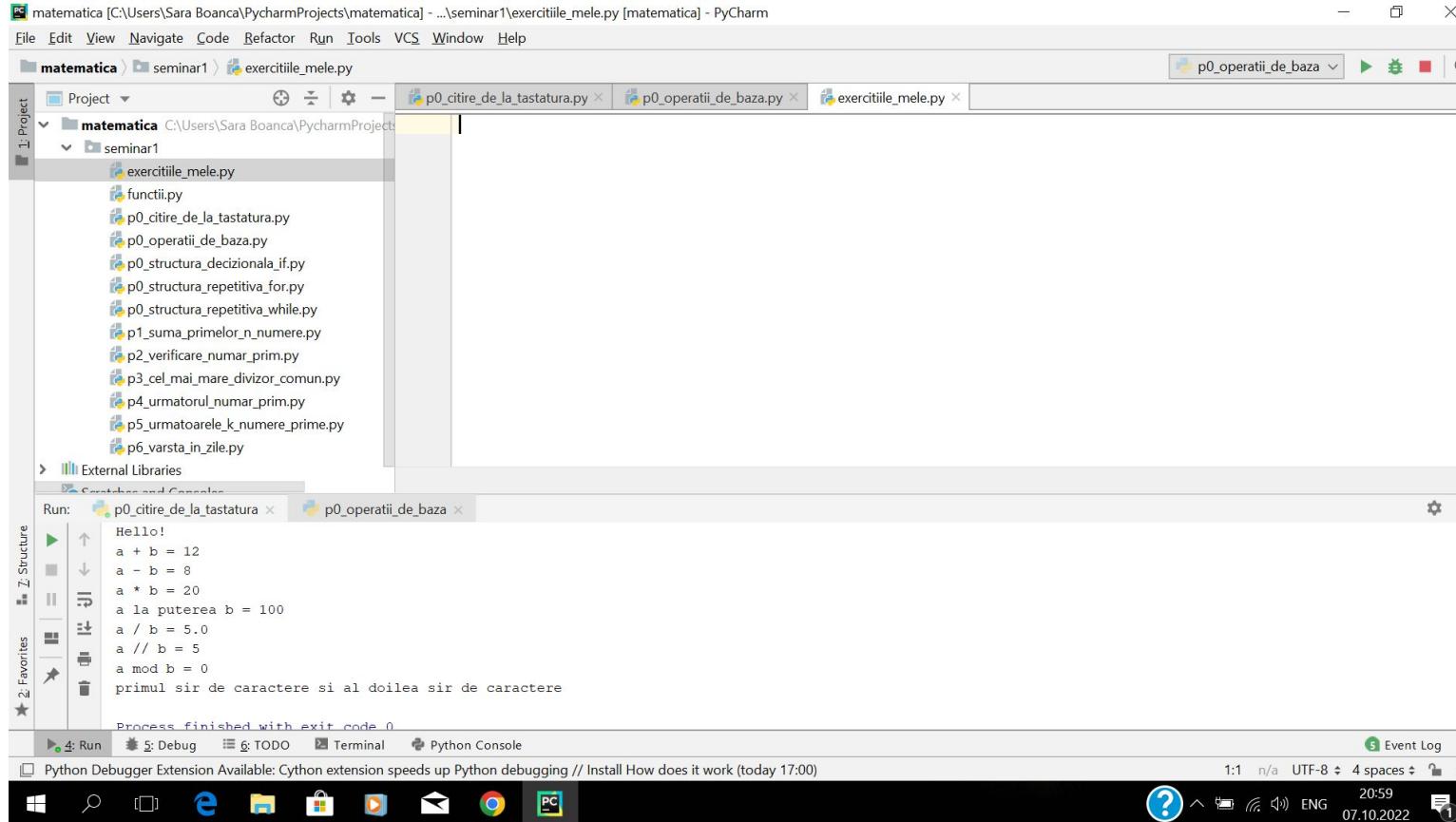
Run: p0\_citire\_de\_la\_tastatura x p0\_operatii\_de\_baza x

Event Log

Python Debugger Extension Available: Cython extension speeds up Python debugging // Install How does it work (today 17:00)

16:1 CRLF UTF-8 4 spaces ENG 20:59 07.10.2022

Daca va uitati in meniul din stanga, acum exercitiile\_mele.py a aparut inauntrul fisierului seminar1. Dati dublu click pe el ca sa il deschideti. Observati ca momentan este gol.



# Scrieti o comanda simpla in fisier.

The screenshot shows the PyCharm IDE interface. The project is named "matematica" and contains a folder "seminar1" which includes several Python files: exercitiile\_mele.py, functii.py, p0\_citire\_de\_la\_tastatura.py, p0\_operatii\_de\_baza.py, p0\_structura\_decizionala\_if.py, p0\_structura\_repetitiva\_for.py, p0\_structura\_repetitiva\_while.py, p1\_suma\_primerelor\_n\_numere.py, p2\_verificare\_numar\_prim.py, p3\_cel\_mai\_mare\_divizor\_comun.py, p4\_urmatorul\_numar\_prim.py, p5\_urmatoarele\_k\_numere\_prime.py, and p6\_varsta\_in\_zile.py. The file "exercitiile\_mele.py" is open in the editor, displaying the code: 

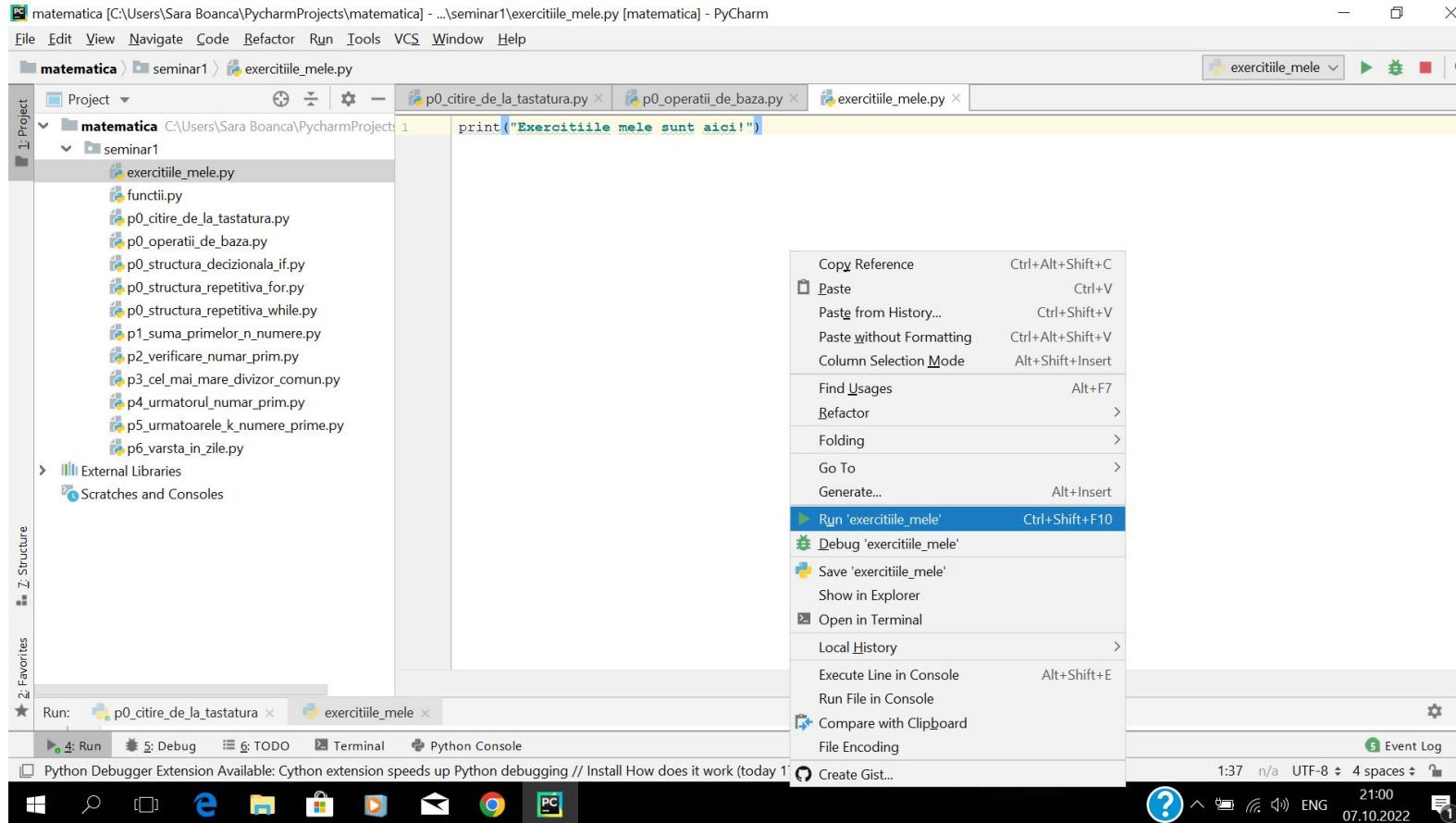
```
print("Exercitiile mele sunt aici!")
```

. The Run tab at the bottom shows the output of the script: 

```
Hello!
a + b = 12
a - b = 8
a * b = 20
a la puterea b = 100
a / b = 5.0
a // b = 5
a mod b = 0
primul sir de caractere si al doilea sir de caractere
```

. The status bar at the bottom right indicates the date and time as 07.10.2022 21:00.

Daca vreti sa rulati codul din fisier, dati click dreapta in fereastra unde este codul si selectati Run. Veti vedea ca mesajul tiparit apare in consola.



Incercati sa rulati exercitiile de la seminar si sa va jucati cu ele.

Pentru Laboratorul 2, va trebui sa rezolvati 2 probleme din lista domnului profesor.

Problemele sunt foarte asemanatoare cu ce am facut la seminar.

Folositi problemele facute impreuna la seminar pentru a rezolva tema.

Tema trebuie predata la laboratorul de saptamana aceasta.

Succes!