MINISTERUL EDUCAȚIEI, CULTURII ȘI CERCETĂRII AL

REPUBLICII MOLDOVA

UNVIERSITATEA LIBERĂ INTERNAȚIONALĂ DIN MOLDOVA

Grupa: C-P-204-11;

Specialitatea: Programarea și analiza produselor de program;

**Raportul**

**„Biletul nr.19”**

****

**Coordonator: Gârlă Eugeniu**

**Elaborat:Vacarciuc Nicolai**

**Chișinău-2022**

**Cuprins**

Capitolul I

Sarcina de lucru………………………………………………………3

Capitol II

1 sarcina…………………………………………………………..4-5

2 sarcina…………………………………………………………..6-7

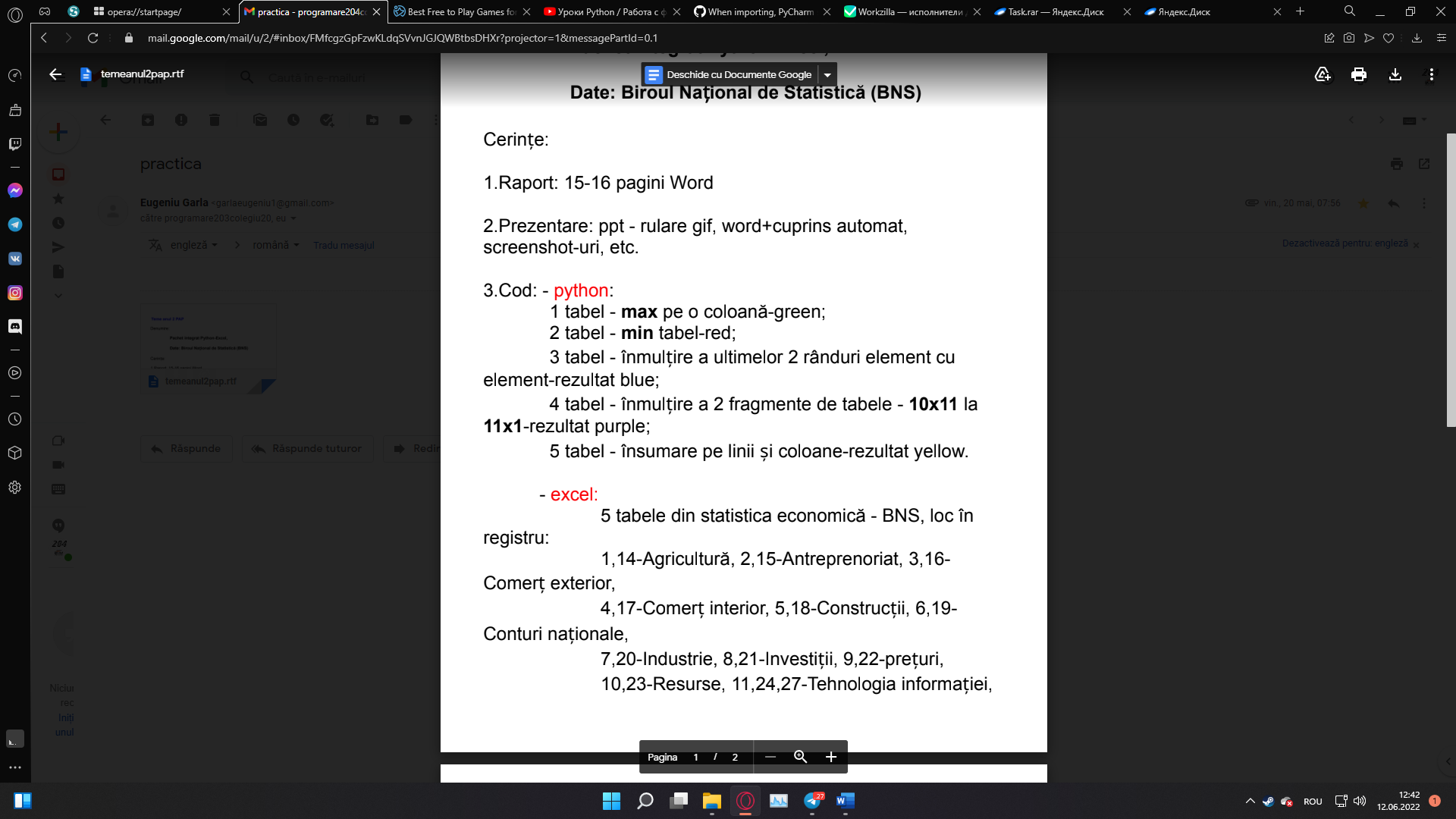
3 sarcina…………………………………………………………..8-9

4 sarcina…………………………………………………………..10-11

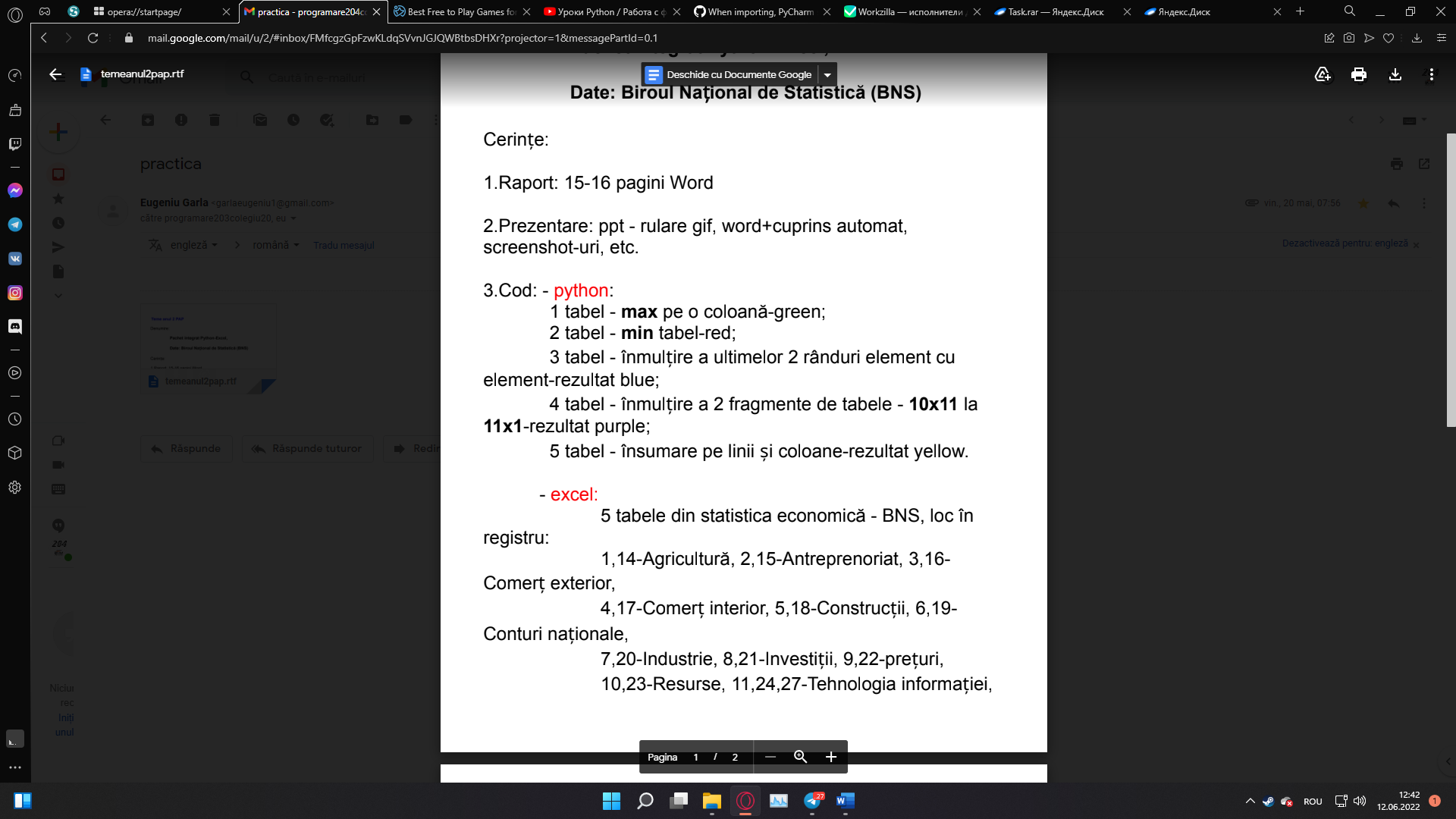
5 sarcina…………………………………………………………..12-13

Concluzii…………………………………………………………………14

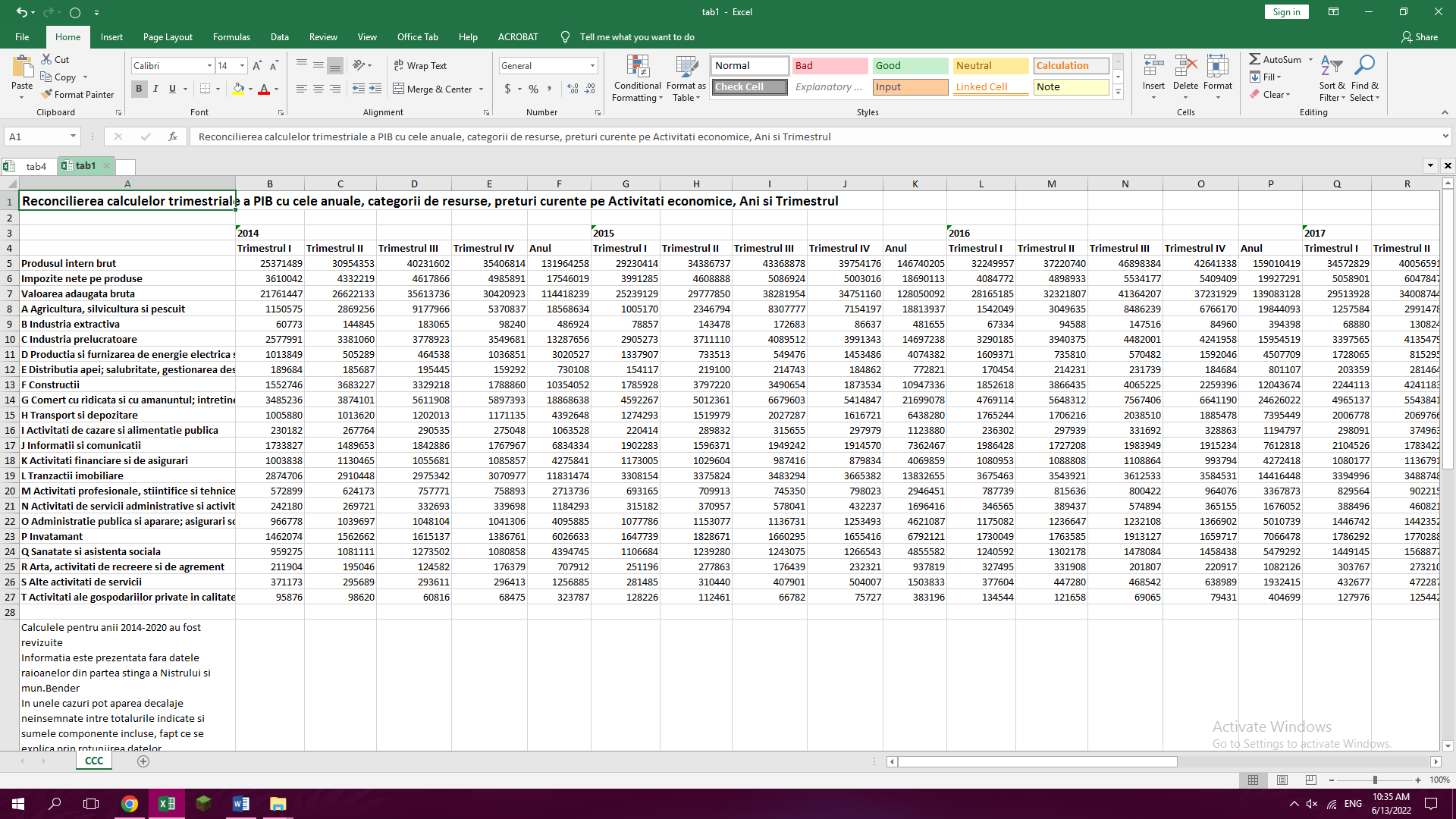
Bibliografie……………………………………………………..………15







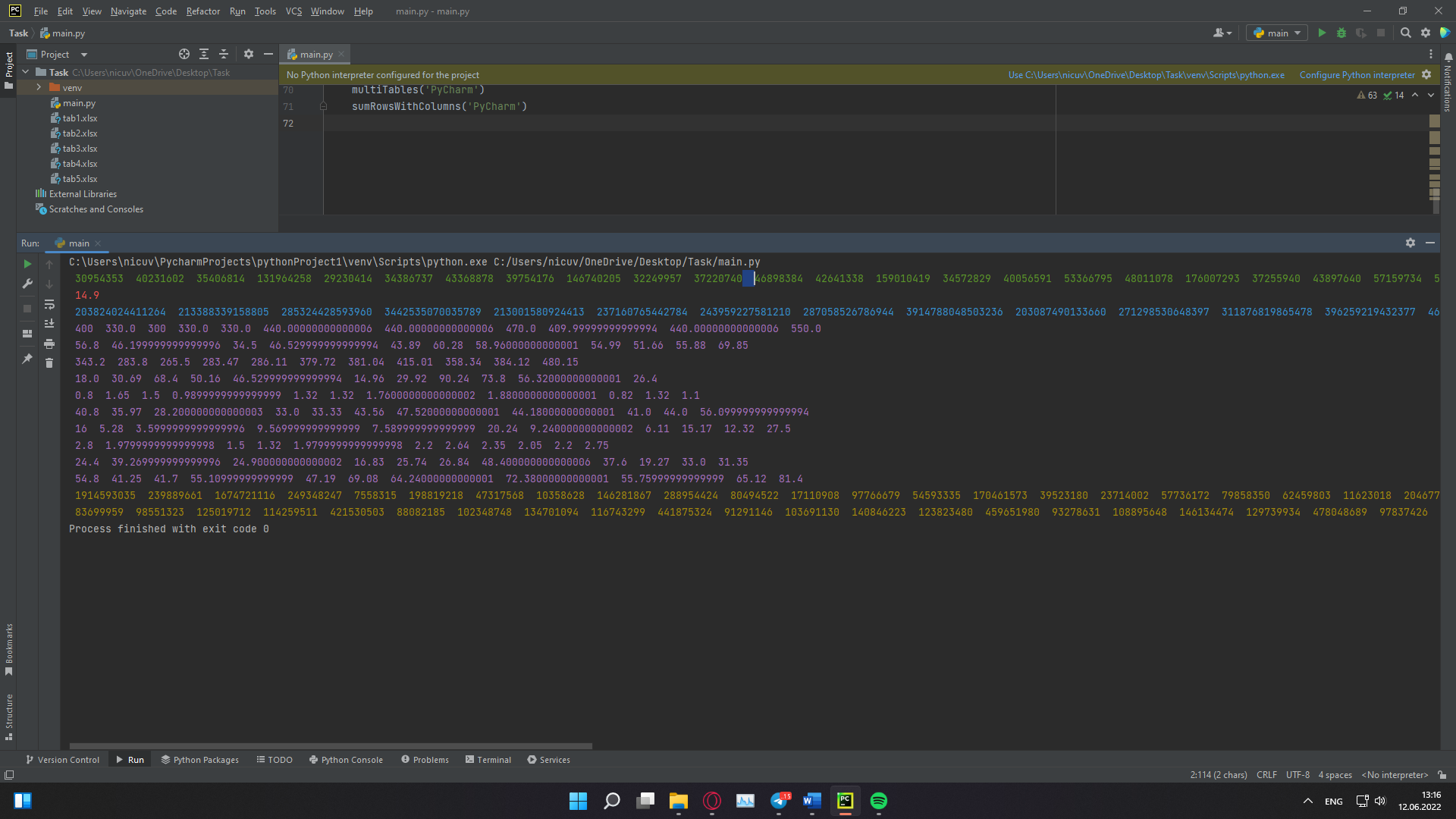
**1 sarcina**

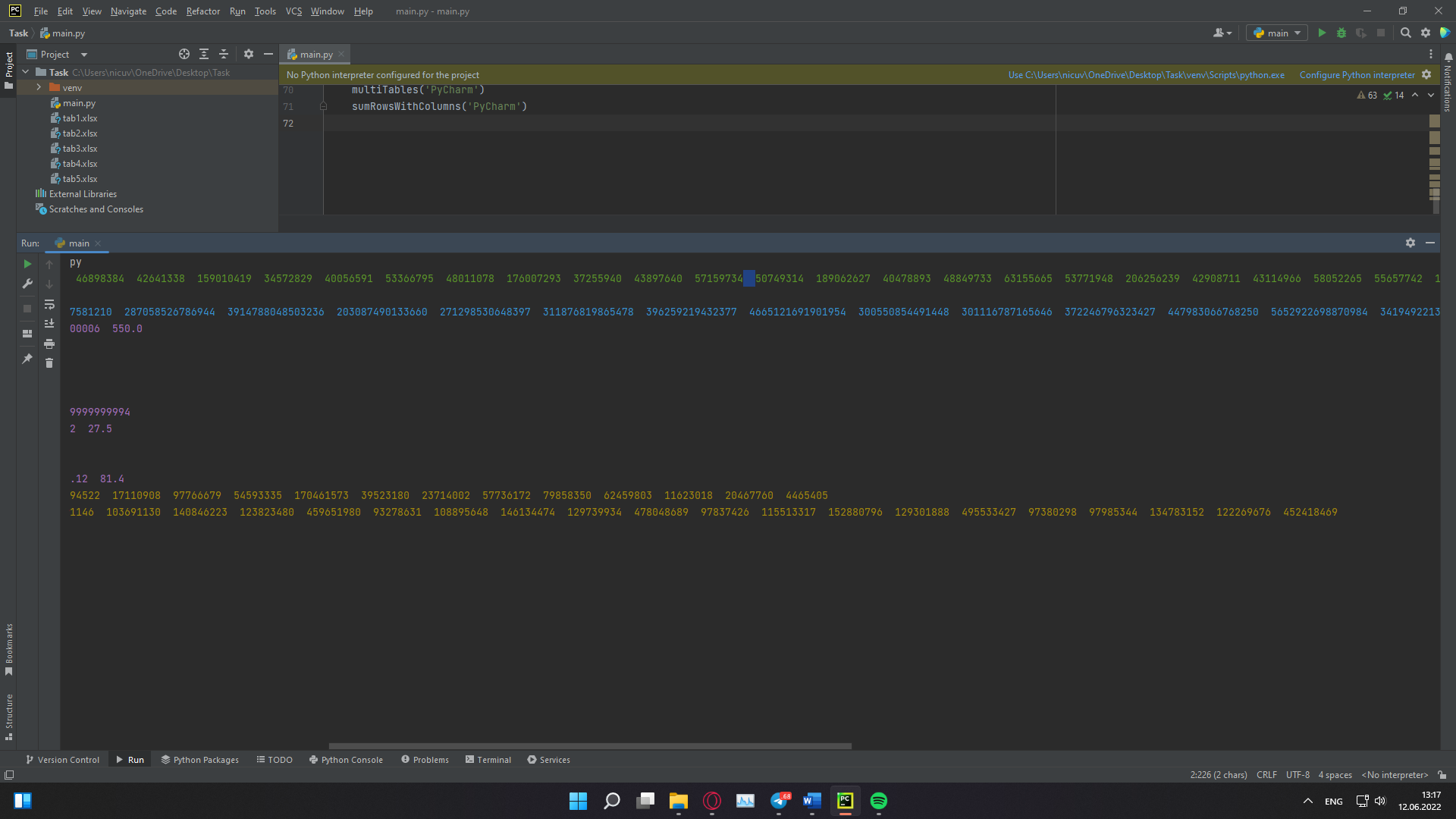


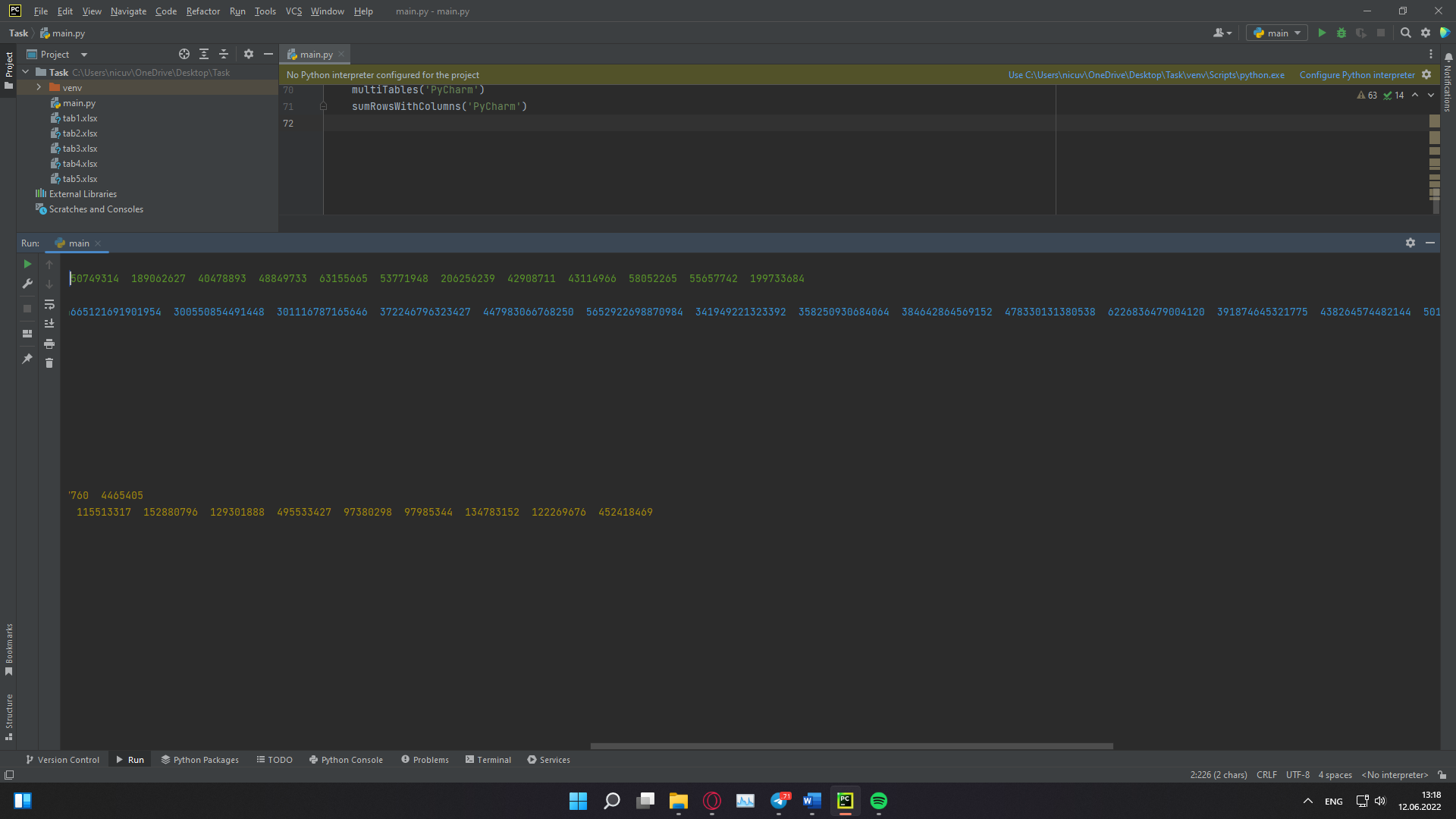
Max pe o coloană-green;

def maxColumnValue(name):  
 book = openpyxl.open("tab1.xlsx", read\_only=True) # citim tabelul în variabilă  
 sheet = book.active # selectăm pagina activă, în cazul nostru este prima  
 for column in range(2, 36): # ciclu pe coloane  
 maxValue = 0 # creem o variabilă pentru a stoca valoarea maximă acolo  
 for row in range(5, 28): # ciclu pe rânduri  
 if (sheet[row][column].value > maxValue): # dacă valoarea este mai mare decât a noastră  
 maxValue = sheet[row][column].value # atunci rescriem variabila  
 print("\033[32m {}" .format(maxValue), end=" ") # afisăm rezultatul cu verde  
 print()

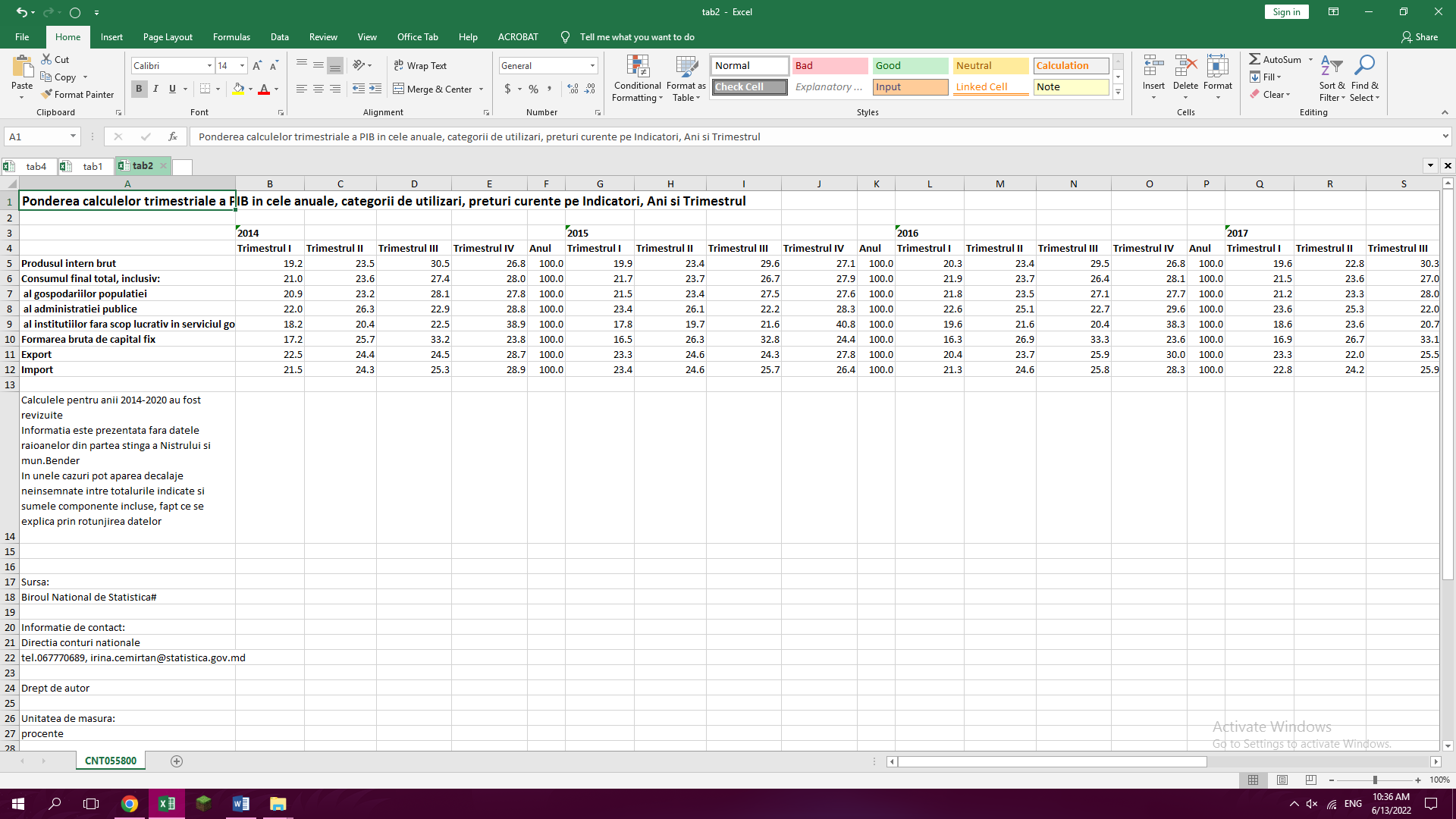
**Datele de ieșire:**







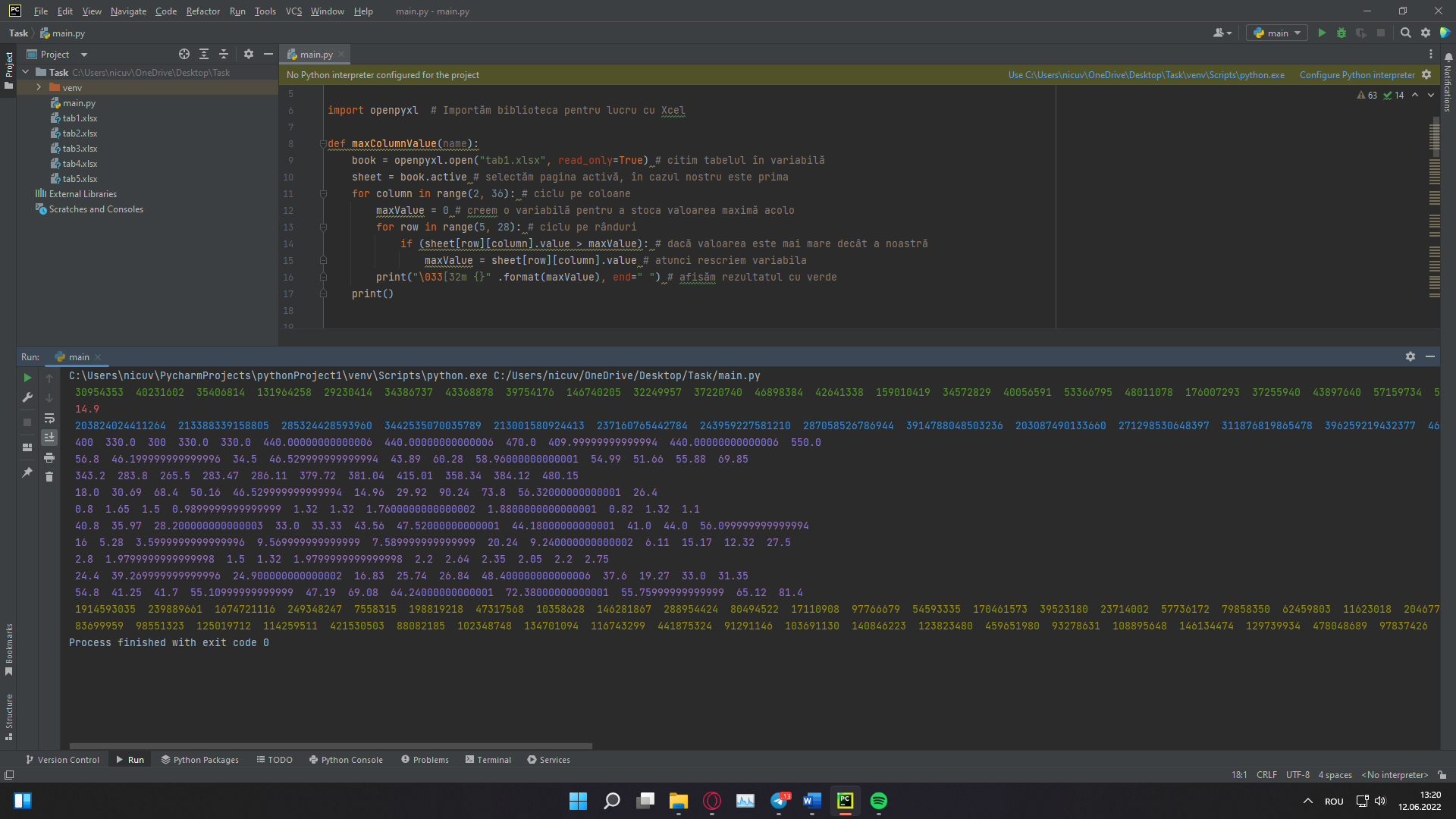
**2 sarcina**



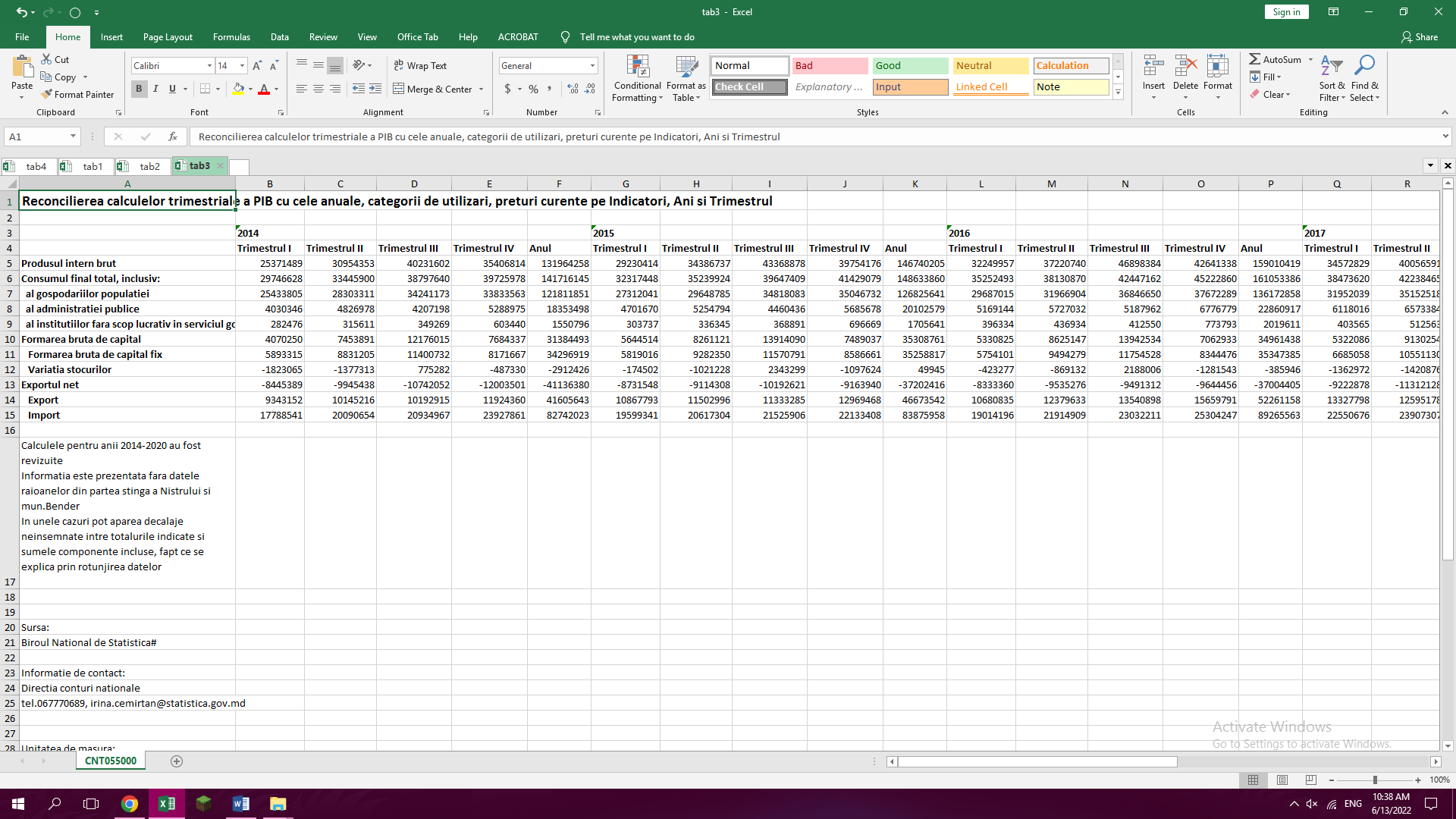
Min table-red;

def minTableValues(name):  
 book = openpyxl.open("tab2.xlsx", read\_only=True) # citim tabelul în variabilă  
 sheet = book.active # selectăm pagina activă, în cazul nostru este prima  
 minValue = 100000 # creem o variabilă pentru a stoca valoarea minimă acolo  
 for column in range(2, 36): # ciclu pe coloane  
 for row in range(5, 13): # cilcu pe randuri  
 if (sheet[row][column].value < minValue): # dacă valoarea este mai mică decât a noastră  
 minValue = sheet[row][column].value # atunci rescriem variabila  
 print("\033[31m {}" .format(minValue)) # afisam rezultatul culoarea rosie

**Datele de ieșire:**



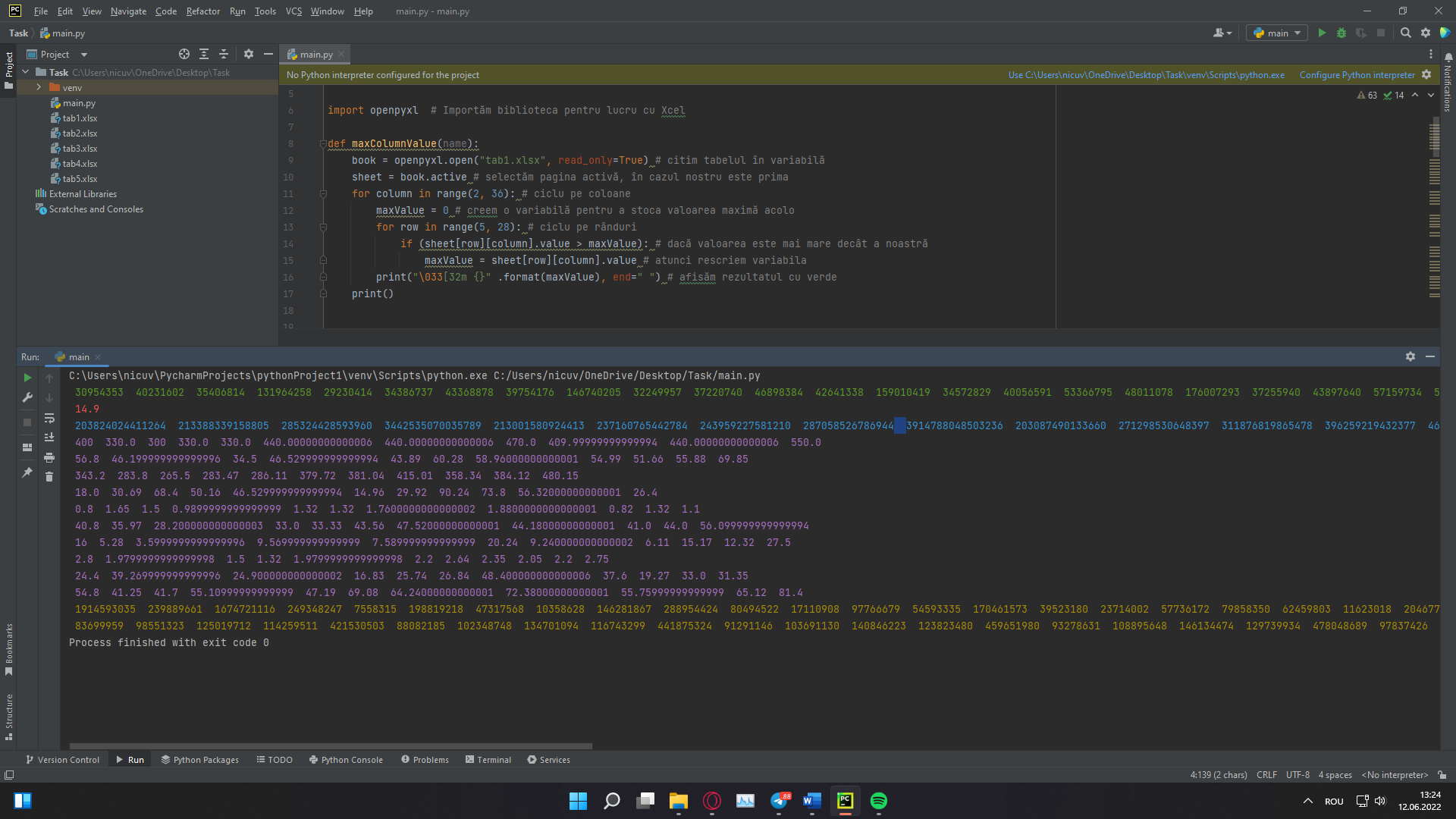
**3 sarcina**

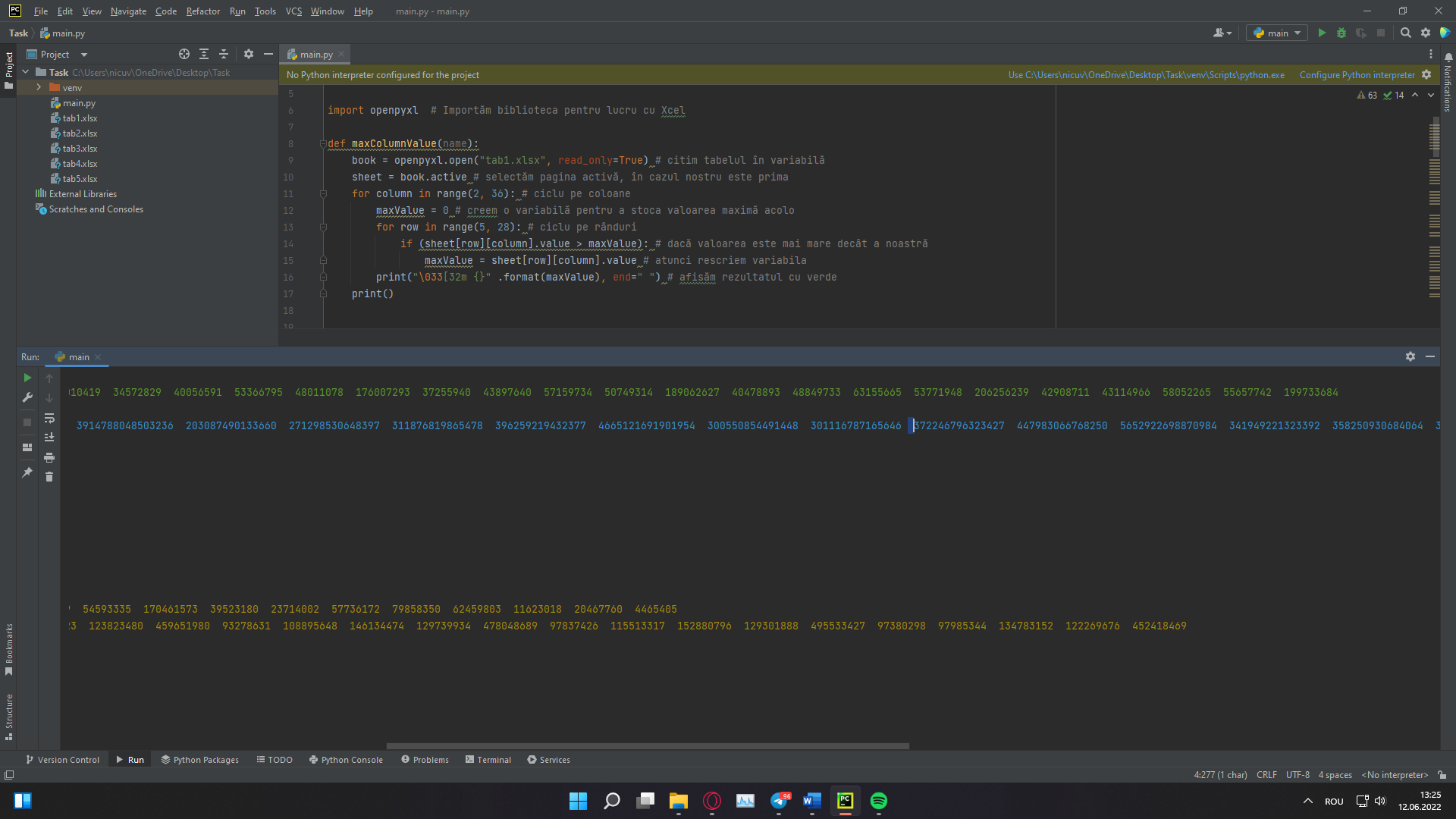


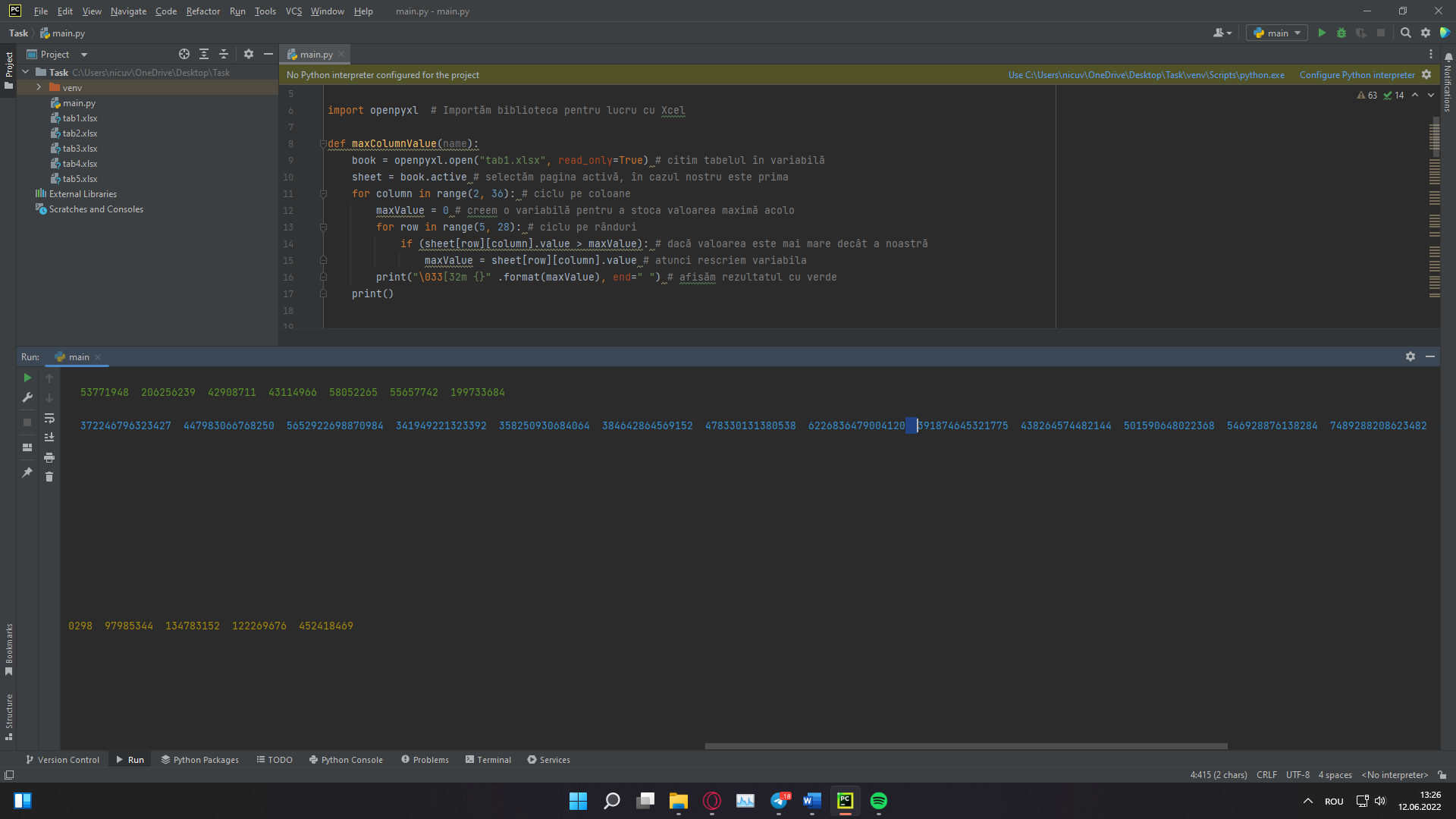
Înmulțire a ultimilor 2 rânduri element cu element-rezultat blue;

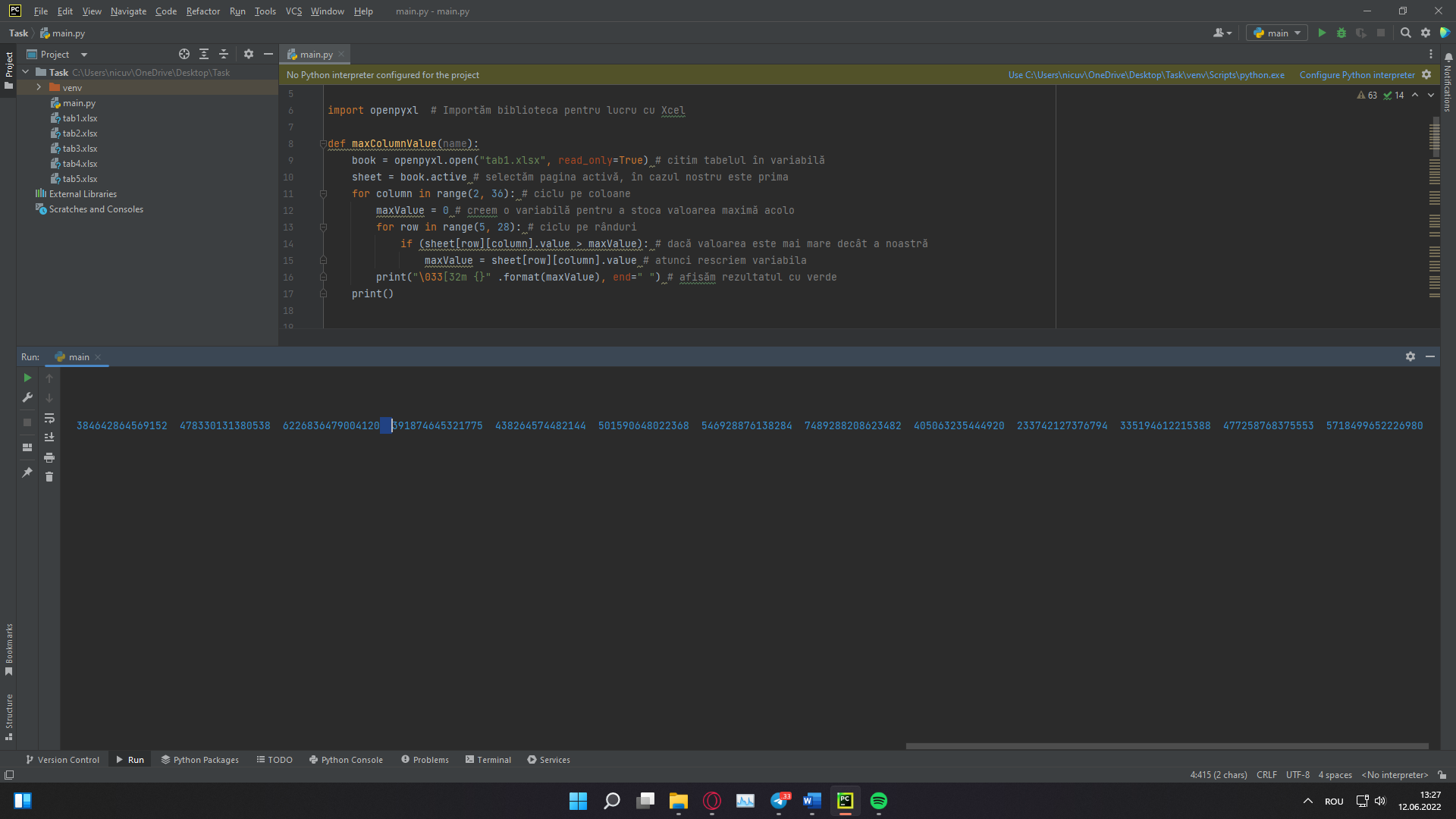
def multiValues(name):  
 book = openpyxl.open("tab3.xlsx", read\_only=True) # citim tabelul în variabilă  
 sheet = book.active # selectăm pagina activă, în cazul nostru este prima  
 for column in range(2, 36): # ciclu pe coloane  
 print("\033[34m {}" .format(sheet[14][column].value \* sheet[15][column].value), end=" ") # înmulțim penultima linie cu ultima și rezultatul în albastru  
 print()

**Datele de ieșire:**

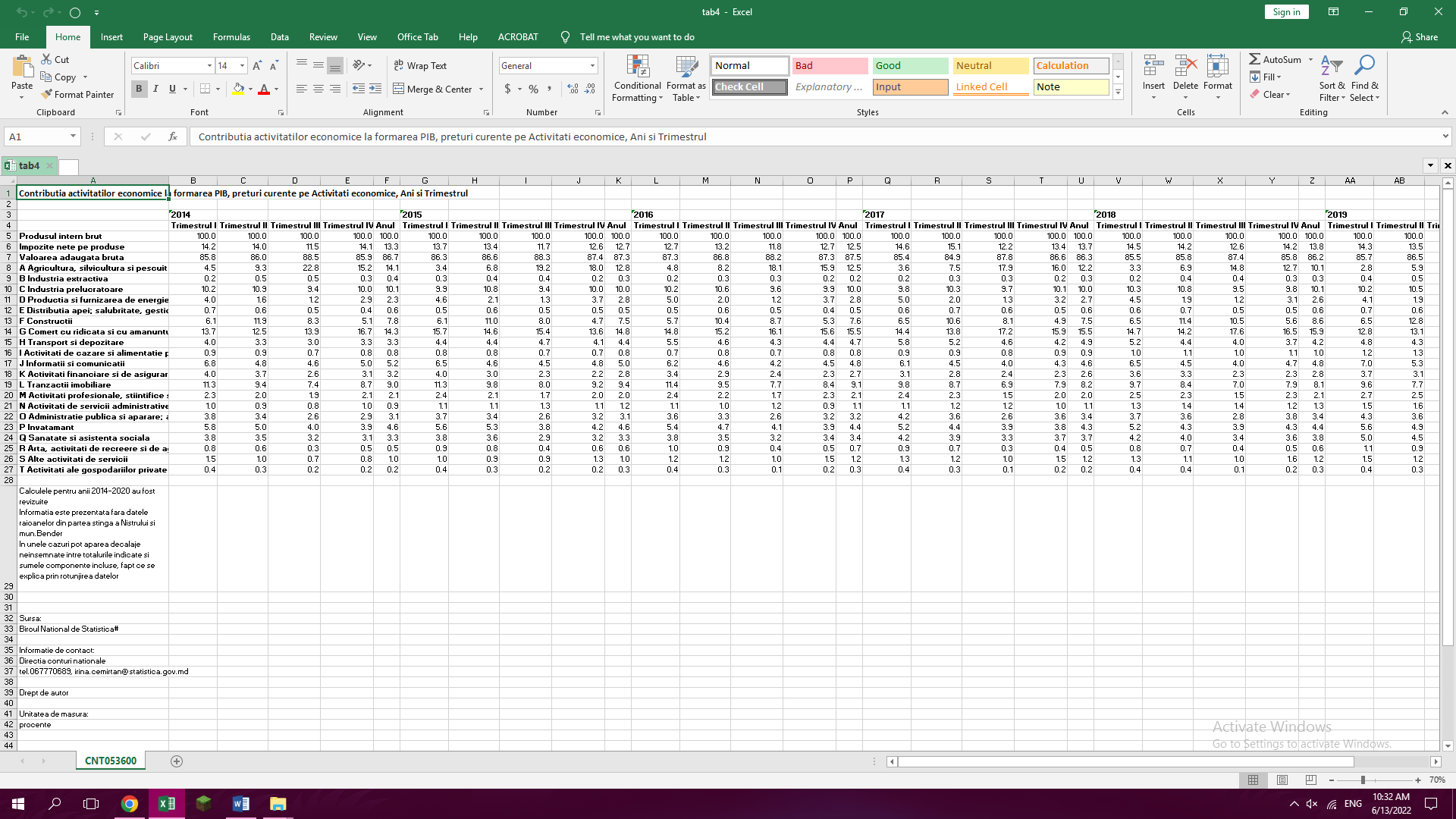








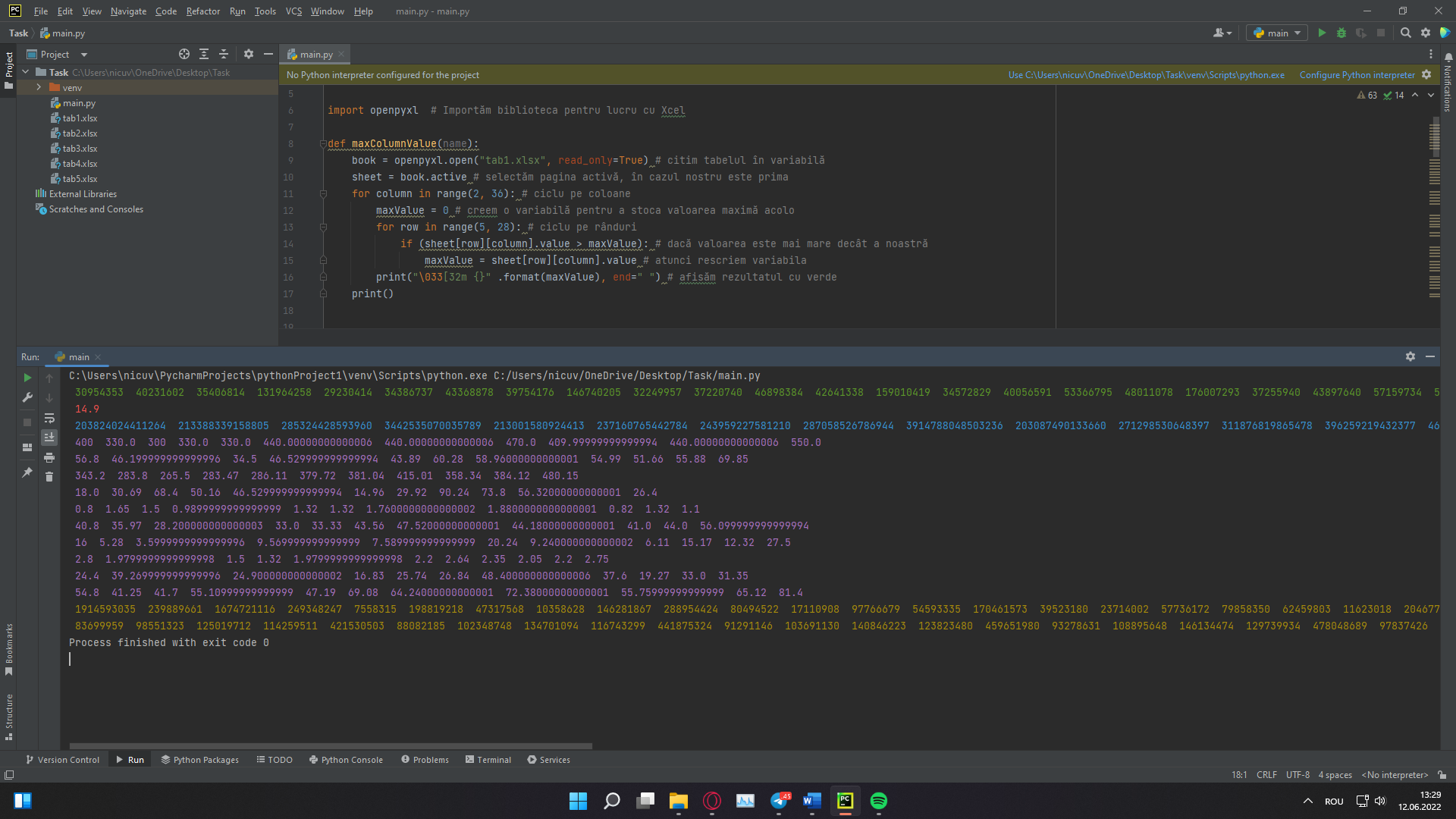
**4 sarcina**



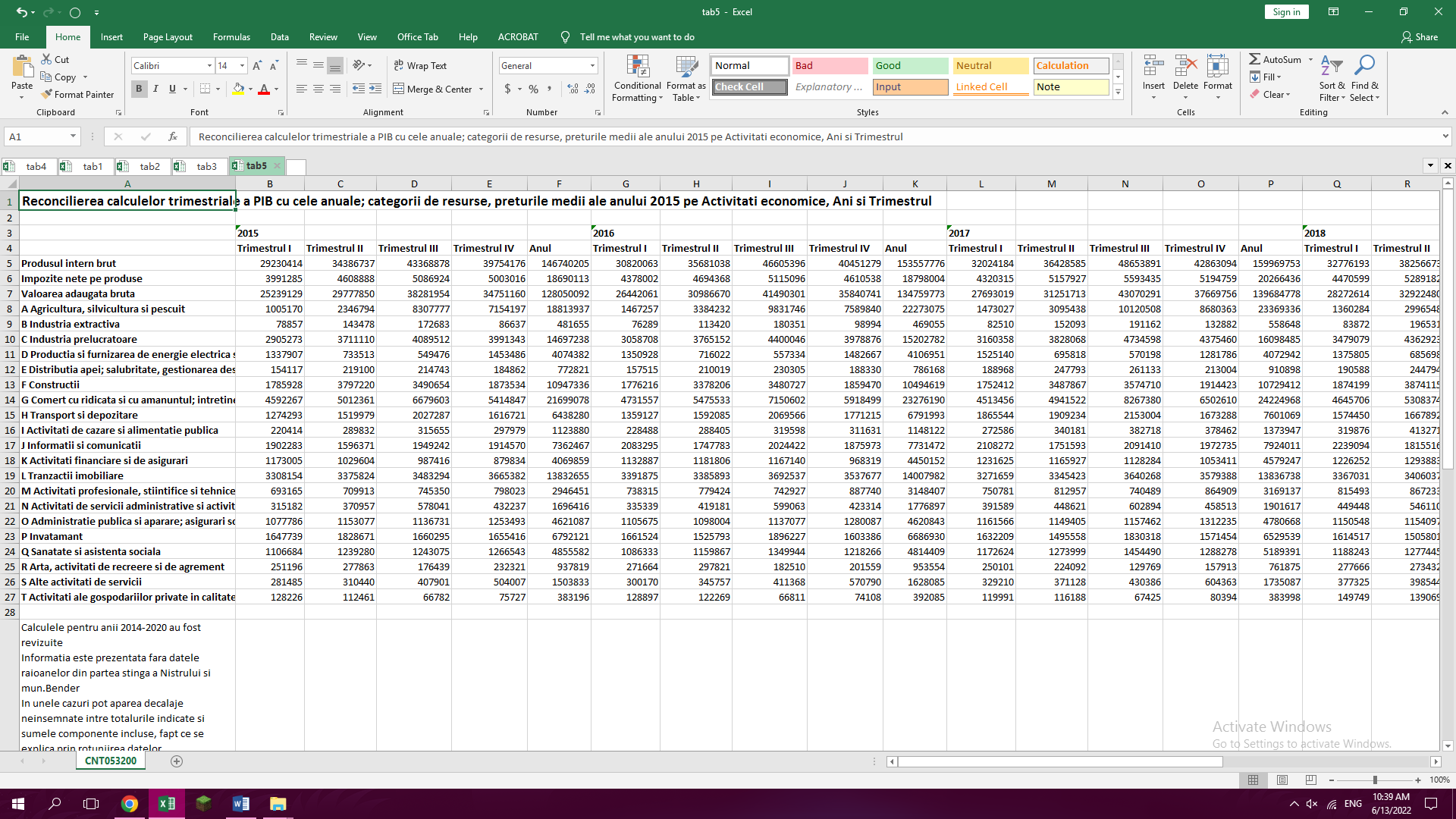
Înmulțire a 2 fragmente de tabele - 10x11 la 11x1-rezultat purple;

def multiTables(name):  
 book = openpyxl.open("tab4.xlsx", read\_only=True) # citim tabelul în variabilă  
 sheet = book.active # selectăm pagina activă, în cazul nostru este prima  
 matrix1 = sheet['B5':'L14'] # alegem prima matrice  
 matrix2 = sheet['B15':'L15'] # alegem a doua matrice  
 for i in range(0, 10): # ciclu pe rânduri  
 for j in range(0, 11): # ciclu pe coloane  
 print("\033[35m {}" .format(matrix1[i][j].value \* matrix2[0][j].value), end=" ") # înmulțim matricele și scoatem rezultatul  
 print()

**Datele de ieșire:**



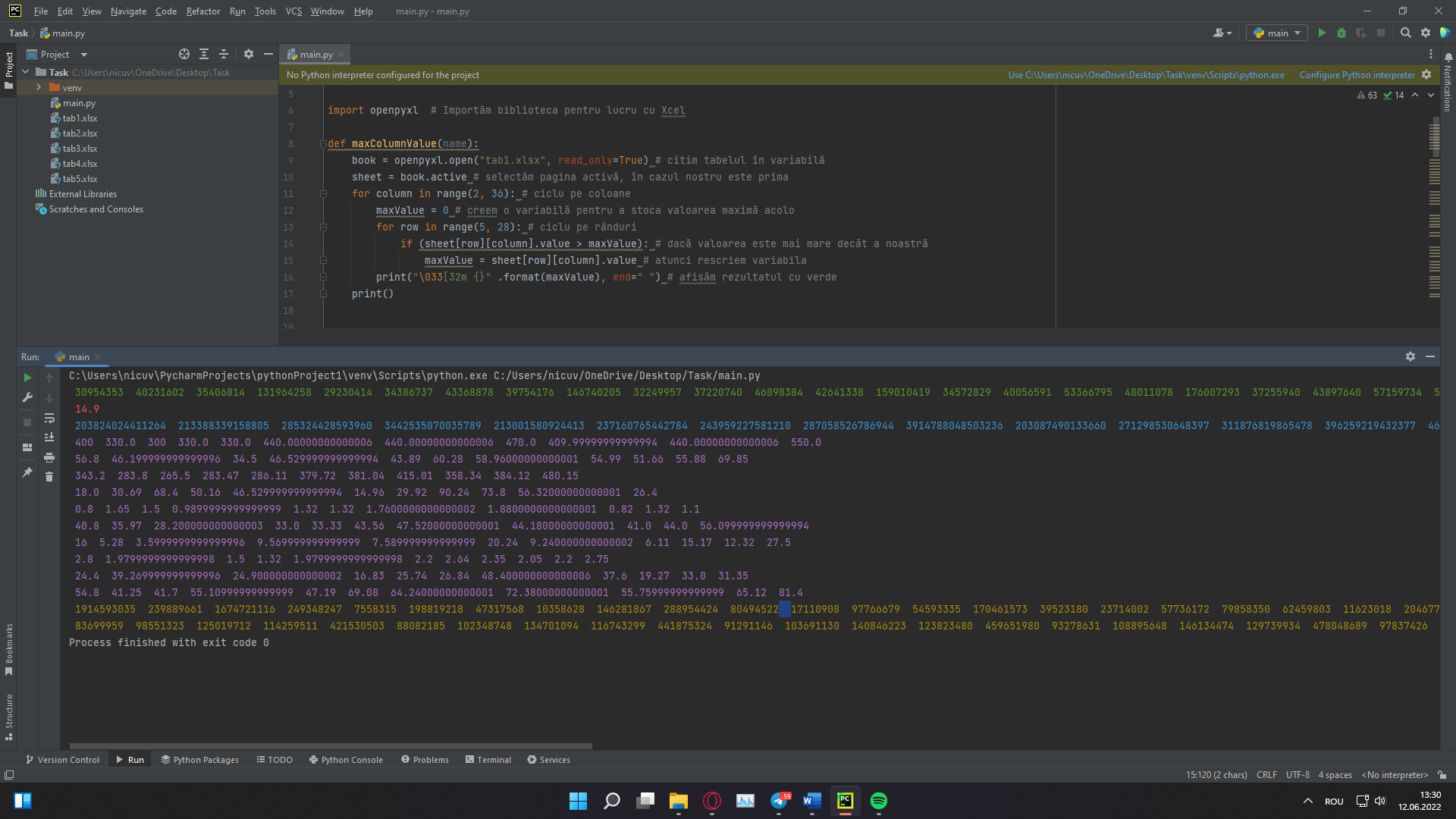
**5 sarcina**

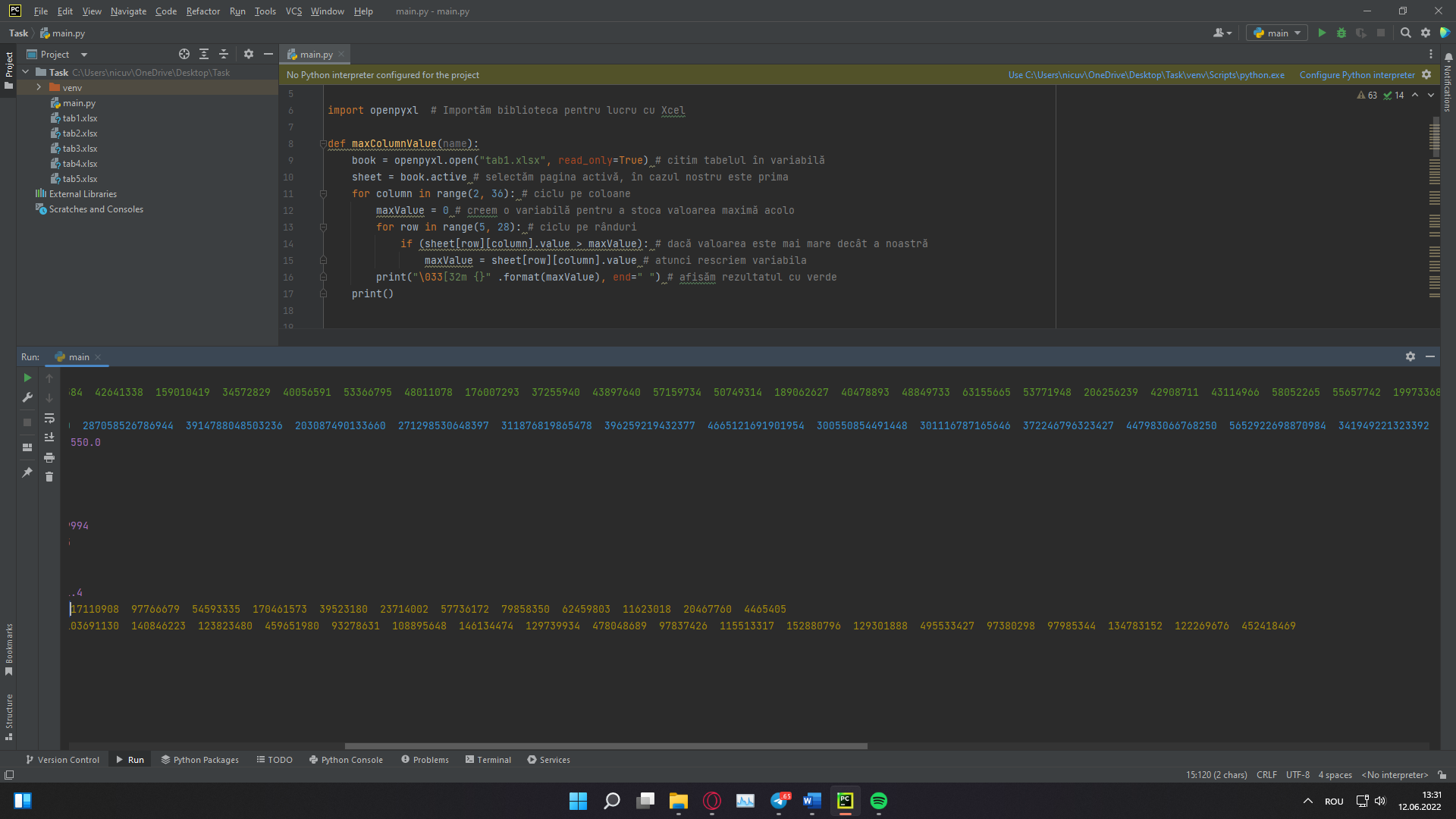


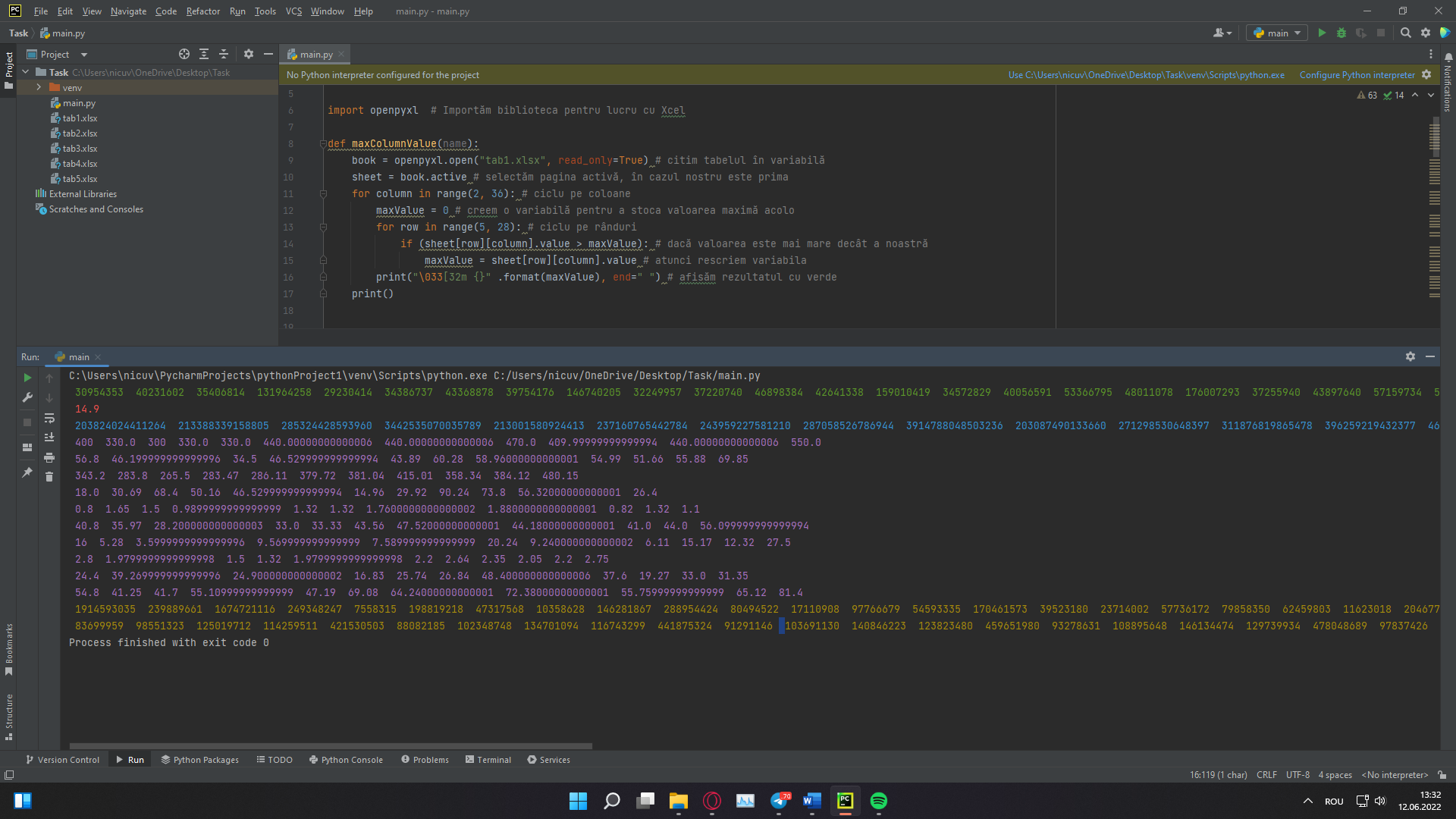
Însumare pe linii și coloane-rezultat yellow.

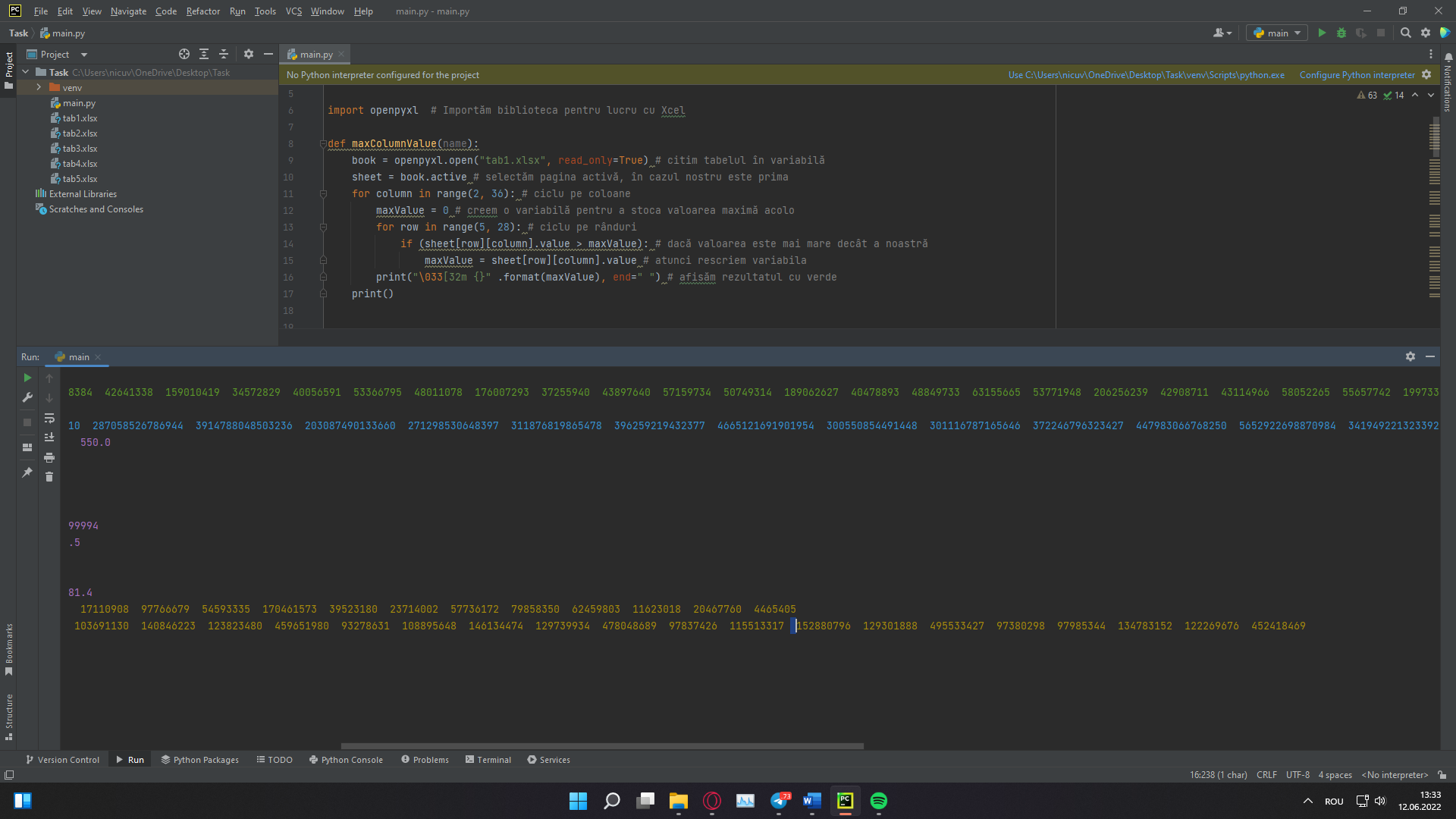
def sumRowsWithColumns(name):  
 book = openpyxl.open("tab5.xlsx", read\_only=True) # citim tabelul în variabilă  
 sheet = book.active # selectăm pagina activă, în cazul nostru este prima  
 matrix = sheet['B5':'AE27'] # selectăm matricea necesară  
 for i in range(0, 23): # ciclu pe rânduri  
 sum = 0 # creem o variabilă în care vom scrie suma și o vom reseta la zero  
 for j in range(0, 30): # ciclu pe coloane  
 sum += matrix[i][j].value # calculăm suma  
 print("\033[33m {}" .format(sum), end=" ") # afișăm sumele pe rânduri în galben  
 print()  
 for i in range(0, 30): # ciclu pe rânduri  
 sum = 0 # creem o variabilă în care vom scrie suma și o vom reseta la zero  
 for j in range(0, 23):# ciclu pe coloane  
 sum += matrix[j][i].value # calculăm suma  
 print("\033[33m {}" .format(sum), end=" ") # afișăm sumele pe coloane în galben

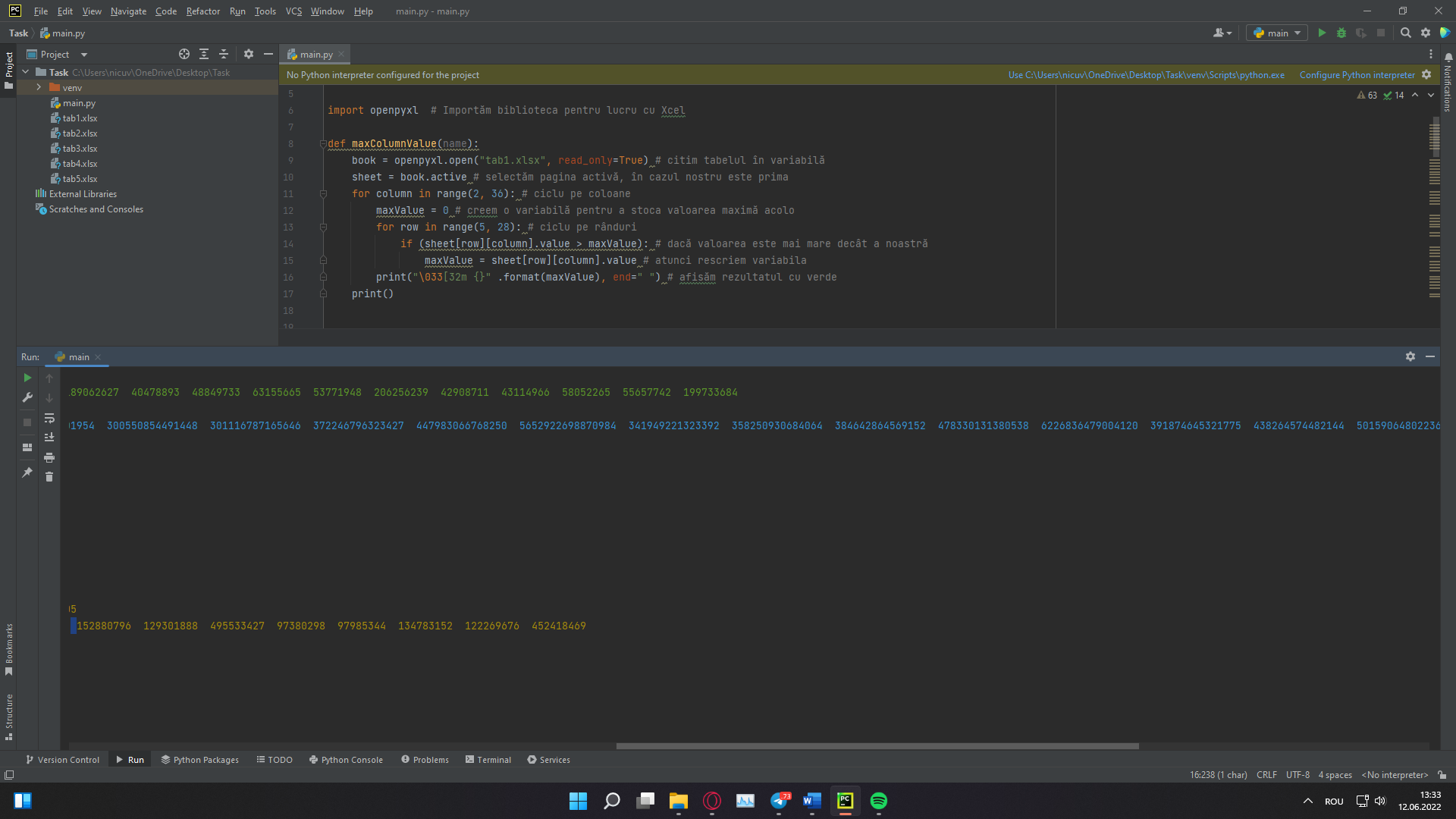
**Datele de ieșire:**

1. 



1. 





**Concluzii**

* **În urma efectuării lucrării practice am înțeles mai detaliat principiul mediului PyCharm pentru limbajul de programare Python, cercetând în internet am efectuat lucrarea la timp.**
* **M-am învățat să mă folosesc și de Excel mai aprofund, învățând a face legătura cu PyCharm.**
* **Activitățile de practică pe care le-am derulat în timpul stagiului de practică au fost pentru mine o experiență venită la momentul propice, ajutându-ma, în special, să dezvolt și să exersez deprinderi ce se bazează pe cunoștințele acumulate în colegiu.**
* **Am descoperit aptitudini pe care nici nu credeam că le posed.**
* **Am interacționat cu baza de date(Excel) și am procesat informații din ea.**
* **M-am învățat să creez și să întroduc date în fișiere.**

**Bibliografie:**

**Datele au fost luate:** **Biroul Național de Statistică (BNS)**

[**https://statistica.gov.md/category.php?l=ro&idc=191**](https://statistica.gov.md/category.php?l=ro&idc=191)

**Toate fișierele pentru examinarea și verificarea lucrului le puteți descarca aici:**

[**https://disk.yandex.ru/d/DIXZp3ieU\_nCvQ**](https://disk.yandex.ru/d/DIXZp3ieU_nCvQ)

**Surse:**

[**https://openpyxl.readthedocs.io/en/stable/#**](https://openpyxl.readthedocs.io/en/stable/)

[**https://www.youtube.com/watch?v=ohfl2QYtWKo**](https://www.youtube.com/watch?v=ohfl2QYtWKo)