PROIECT GRAFIC PERFORMANTA CALCULATOARELOR

Strat Bogdan, Avram Miruna

Proiectul este reprezentat de un script python care:

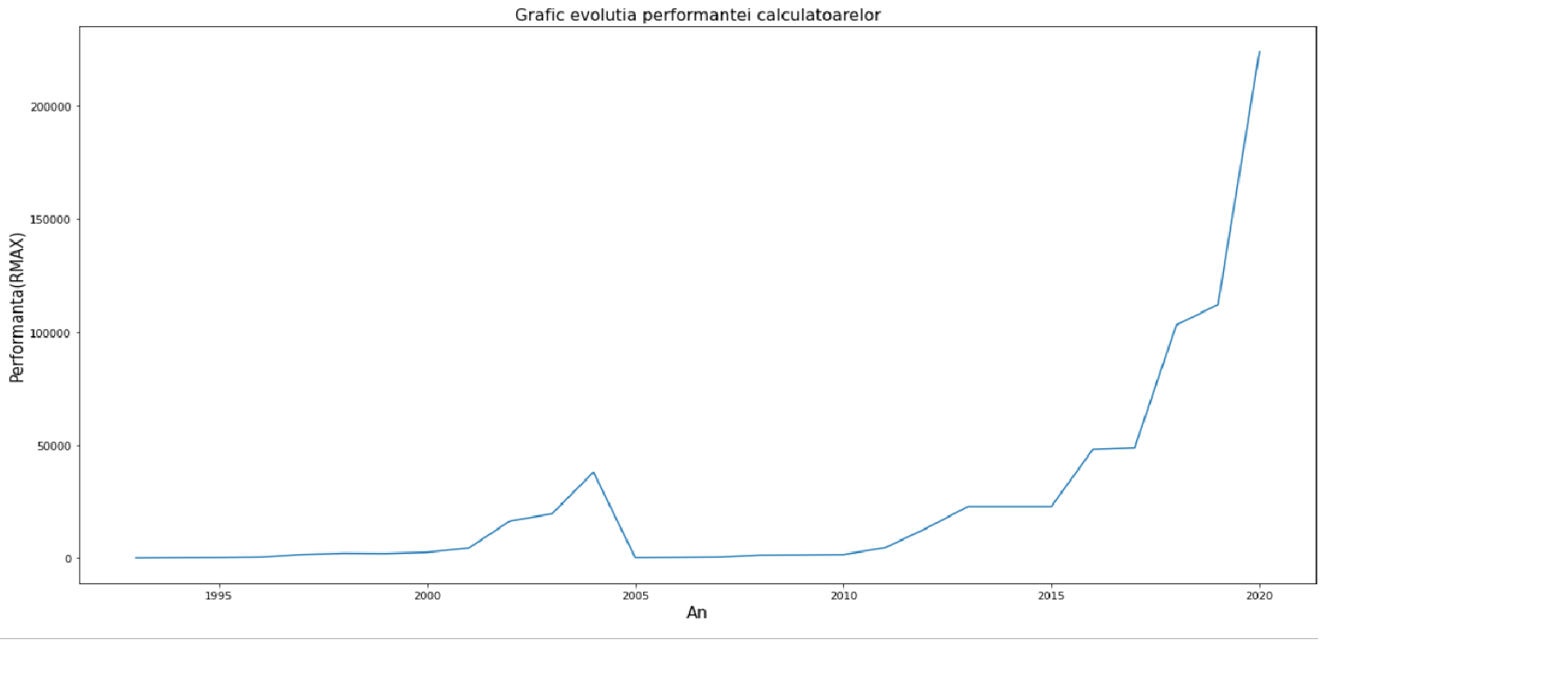
1. scaneaza https://www.top500.org/
2. gaseste pe fiecare an primele 3 cele mai puternice sisteme
3. realizeaza media pe luna/an
4. afiseaza o figura cu evoluția performantei vs. timp
5. calculeaza progresul mediu realizat pe an

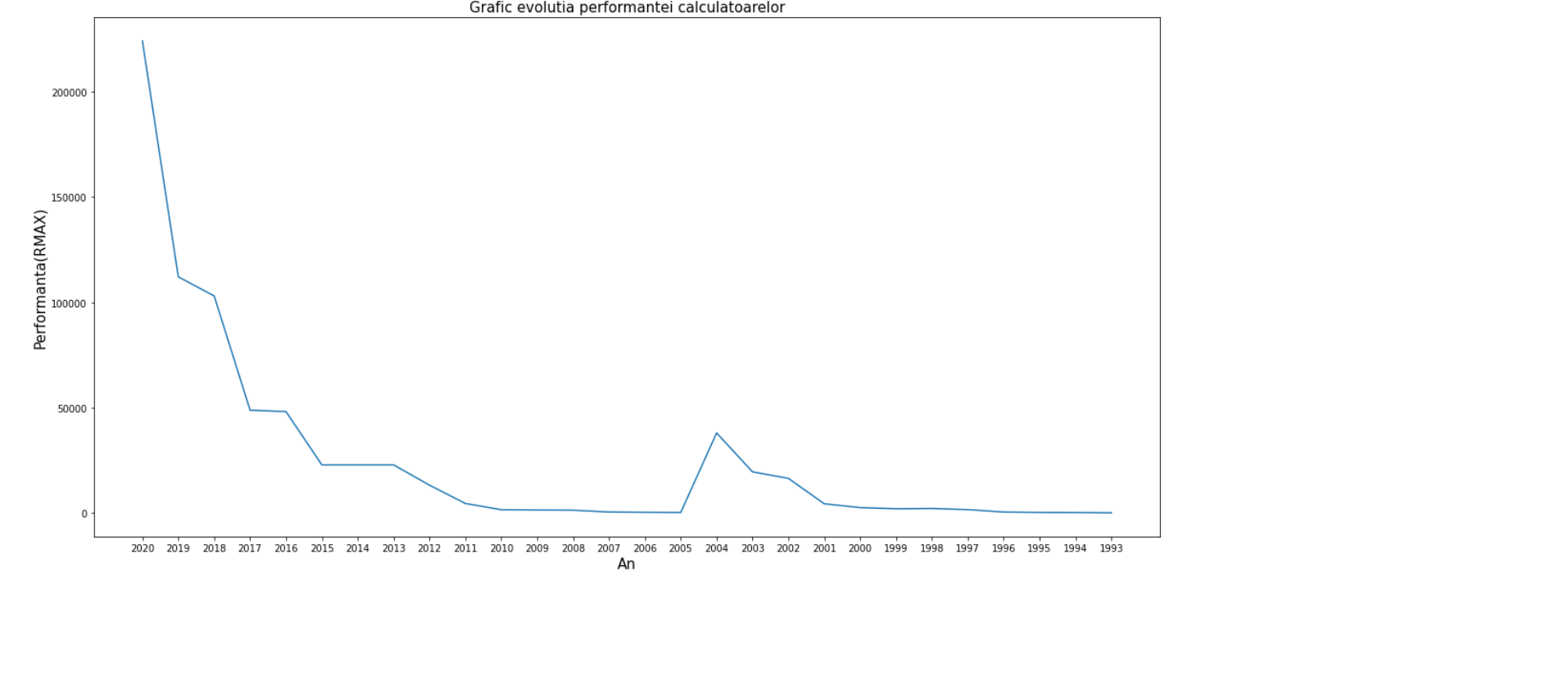
Pasi proiect:

* scanarea https://www.top500.org/ , utilizarea metodei find\_all cu scopul de a gasi toate link-urile de pe pagina respectiva , memorarea in *containers* tuturor ul-urilor (unordered list) care au id-ul "squarelist"
* cum ‘a’ este tag-ul pentru link din html, cautam fiecare link din lista cu ani din site-ul suport si cream link-ul fiecarei luni, daca este posibila accesarea acelui link ( daca pagina respectiva functioneaza)
* daca se poate accesa pagina unei anumite luni, cautam eticheta ‘td’ si obtinem manual valorile Rmax ale primelor 3 sisteme, calculand ,totodata, si media lor atat pe luna (cand contor = 1), cat si pe an ( cand contor = 2)
* retinem intr-un array valorile performantei fiecarui an ( axa\_performanta )
* retine intr-un array fiecare an ( axa\_an)
* in functie de cele 2 array-uri cream diagrama in python
* calcuam progresiul mediu pe fiecare an si facem media aritmetica a rezultatului obtinut

Link github:

Grafic evolutie:





**Progresul mediu este: 1.626000962064051**

Codul proiectului:

#!/usr/bin/python3.6

import http.client

import bs4 as bs

import urllib.request

from matplotlib import pyplot as plt

site="www.top500.org"

site\_liste="https://" +site+"/lists/top500"

source=urllib.request.urlopen(site\_liste).read()

soup=bs.BeautifulSoup(source,'lxml')

#accesez pagina cu toate topurile

containers=soup.find\_all('ul',{"id":"squarelist"})

years\_page=containers[0]

verificare\_exceptie=1

k=0

medie\_an=0

contor\_luna=0

contor\_an=0

an1=2020

an2=2019

contor\_scadere\_an=0

axa\_an=[]

axa\_performanta=[]

contor1=0

contor\_td=1

eroare\_prima\_luna=0

#caut linkul fiecarei pagini care are un top

for url in years\_page.find\_all('a'):

verificare\_exceptie=1

#creez linkul fiecarei pagini care are top

link=site\_liste+"/"+url.get('href')

k=k+1

contor1=contor1+1

if contor1==2:

sir\_an=url.get('href')[0:4]

axa\_an.append(int(sir\_an))

contor1=0

try:

urllib.request.urlopen(link).read()

except:

verificare\_exceptie=0

if verificare\_exceptie==0:

if contor\_luna==0:

eroare\_prima\_luna=1

elif contor\_luna==1:

medie\_an=medie\_luna

contor\_luna=0

axa\_performanta.append(medie\_an)

medie\_an=0.0

if verificare\_exceptie==1:

source1=urllib.request.urlopen(link).read()

soup1=bs.BeautifulSoup(source1,'lxml')

tds=soup1.find\_all('td')

#retin doar td-urile primelor 3 clasate

tds=tds[:18]

contor\_rank=0

nr=0

ok=0

medie\_luna=0.0

for td in tds:

#selectez si modifc in int rmax-ul primelor 3 clasate

if contor\_rank==3 and ok==0:

aux=str(td)

aux=aux[31:(len(aux)-5)]

if aux.find(',')==-1:

pass

else:

aux=aux.split(',')

nr=int(aux[0])\*1000

aux=aux[1]

aux=aux.split('.')

nr=nr+int(aux[0])

aux=aux[1]

nr=nr+(int(aux)/10)

medie\_luna=medie\_luna+nr

contor\_rank=0

ok=1

elif ok==0:

contor\_rank=contor\_rank+1

if contor\_rank==6 and ok==1:

aux=str(td)

aux=aux[31:(len(aux)-5)]

if aux.find(',')==-1:

pass

else:

aux=aux.split(',')

nr=int(aux[0])\*1000

aux=aux[1]

aux=aux.split('.')

nr=nr+int(aux[0])

aux=aux[1]

nr=nr+(int(aux)/10)

medie\_luna=medie\_luna+nr

#print(medie\_luna)

contor\_rank=0

ok=1

contor\_rank=1

elif ok==1:

contor\_rank=contor\_rank+1

#calculez media pe luna,respectiv pe an

contor\_luna=contor\_luna+1

medie\_luna=medie\_luna/3

if eroare\_prima\_luna==1:

medie\_an=medie\_luna

axa\_performanta.append(medie\_an)

medie\_an=0.0

contor\_luna=0

eroare\_prima\_luna=0

elif eroare\_prima\_luna==0:

medie\_an=medie\_an+medie\_luna

if contor\_luna==2:

medie\_an=medie\_an/2

axa\_performanta.append(medie\_an)

contor\_luna=0

medie\_an=0.0

an1=2020

an2=2019

progres\_mediu=0

for i in range(len(axa\_performanta)-1):

an1=an1-1

an2=an2-1

progres\_mediu=progres\_mediu+(axa\_performanta[i]/axa\_performanta[i+1])

progres\_mediu=progres\_mediu/len(axa\_performanta)

print("Progresul mediu este: ",progres\_mediu,end='\n')

#Plotting to our canvas

fig=plt.figure(figsize=(20,10))

plt.plot(axa\_an,axa\_performanta)

plt.title('Grafic evolutia performantei calculatoarelor',fontsize=15)

plt.ylabel('Performanta(RMAX)',fontsize=15)

plt.xlabel('An',fontsize=15)

#Showing what we plotted

plt.show()