



IPAQ PETA

V. GIMNAZIJA ZAGREB

GIMNAZIJA VUKOVAR

SREDNJA ŠKOLA LOVRE MONTIJA KNIN

SREDNJA ŠKOLA PAKRAC

GIMNAZIJA METKOVIĆ

PMF ZAGREB

Višeprocorsko programiranje

ULAGANJE U BUDUĆNOST



PROJEKT
SUFINANCIRA
EUROPSKA UNIJA

Ova publikacija je izrađena uz pomoć Europske unije. Sadržaj publikacije je u isključivoj nadležnosti V. gimnazije te se ni na koji način ne može smatrati da odražava stajališta Europske unije.

Europsku uniju čini 28 država članica koje su odlučile postupno povezivati svoja znanja, resurse i sudbine. Tijekom 50-godišnjeg razdoblja proširivanja, zajedno su izgradile područje stabilnosti, demokracije i održivog razvoja, zadržavajući pritom kulturnu raznolikost, toleranciju i slobode pojedinaca. Europska unija je posvećena dijeljenju svojih postignuća i vrijednosti s državama i narodima izvan svojih granica.

Procesi

- procesi se mogu izvoditi na različitim računalima (ali i na istom)
- dretve se uvijek izvode na jednom računalu
- dretve dijele isti memorijski prostor, dok procesi imaju svaki svoj prostor
- modul **multiprocessing**
- klasa **Process**
- parametri konstruktora:
 - **target** – metoda koja će se izvoditi unutar dretve
 - **args** – parametri metode (navode se unutar okruglih zagrada)
- metode:
 - **start()** – pokreće dretvu
 - **join()** – osigurava da će glavni program čekati dok se odgovarajuća dretva ne izvede do kraja
- zgodna funkcija:
 - **cpu_count()** – vraća broj jezgri (procesora) računala

ULAGANJE U BUDUĆNOST



PROJEKT
SUFINANCIRA
EUROPSKA UNIJA



Primjer 5:

Ispišimo sve proste brojeve od 100000000000 do 10000020000.

ULAGANJE U BUDUĆNOST



PROJEKT
SUFINANCIRA
EUROPSKA UNIJA



Primjer 5 - Rješenje:

```
def prost(n):  
    for i in range(2, round(n ** 0.5 + 1)):  
        if n % i == 0:  
            return False  
    return True  
  
n = 100000000000  
m = 10000020000  
for i in range(n, m + 1):  
    if prost(i):  
        print(i)
```

- Intel Core i5 750 @2.67GHz
- 4 jezgre
- 4GB
- vrijeme izvođenja: **21.35s**

ULAGANJE U BUDUĆNOST



PROJEKT
SUFINANCIRA
EUROPSKA UNIJA



Primjer 5 - Rješenje:

```
from multiprocessing import *
from time import *

def prost(n):
    for i in range(2, round(n ** 0.5 + 1)):
        if n % i == 0:
            return False
    return True

def prosti(a, b):
    for i in range(a, b + 1):
        if prost(i):
            print(i)
    return

if __name__ == '__main__':
    start = clock()
    n = 100000000000
    m = 10000020000
    k = 2
    d = (m - n) // k
    procesi = [Process(target = prosti, args
        = (n + i * d, n + (i + 1) * d, )) for
        i in range(k)]
    for p in procesi:
        p.start()
    for p in procesi:
        p.join()
    print("%f sekundi" % (clock() - start))
```

- vrijeme izvođenja: **6.85s**
- k = 2: **11.87s**

Pitanja za razmišljanje?

- kako prebrojati proste brojeve unutar nekog intervala pomoću višeprocorskih programa?
 - što ako dva procesa istovremeno pokušavaju pisati u neku varijablu ("race condition")?
- Kako napraviti ravnopravniju podjelu intervala?
- Što ako jedan proces treba podatke od drugog procesa koji još nije gotov? Što ako drugi proces ovisi ponovo o prvom procesu ("deadlock")?

ULAGANJE U BUDUĆNOST



PROJEKT
SUFINANCIRA
EUROPSKA UNIJA

