Proiect SSI

joi, 4 ianuarie 2024 11:27

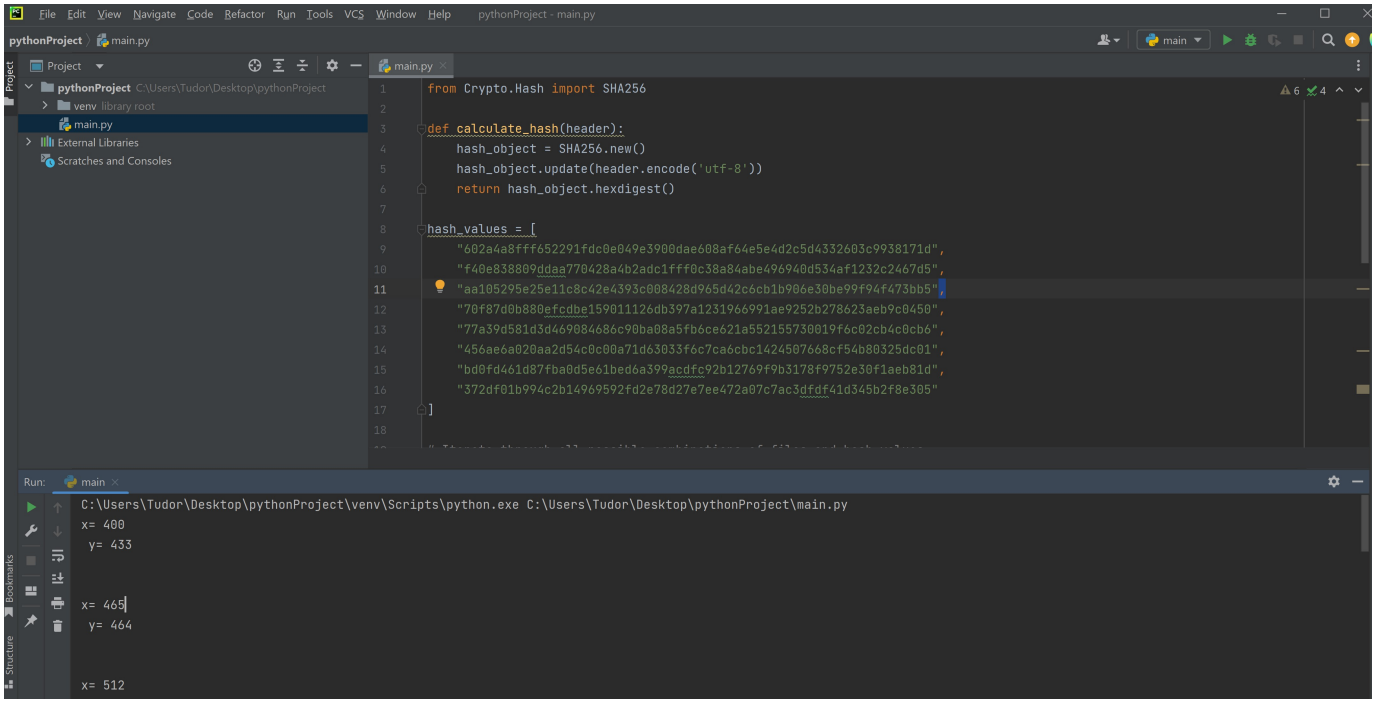
```
fromCrypto.HashimportSHA256

defcalculate_hash(header):
hash_object=SHA256.new()
hash_object.update(header.encode('utf-8'))
returnhash_object.hexdigest()

hash_values=[
"602a4a8fff652291fdc0e049e3900dae608af64e5e4d2c5d4332603c9938171d",
"f40e838809ddaa770428a4b2adc1fff0c38a84abe496940d534af1232c2467d5",
"aa105295e25e11c8c42e4393c008428d965d42c6cb1b906e30be99f94f473bb5",
"70f87d0b880efcdbe159011126db397a1231966991ae9252b278623aeb9c0450",
"77a39d581d3d469084686c90ba08a5fb6ce621a552155730019f6c02cb4c0cb6",
"456ae6a020aa2d54c0c00a71d63033f6c7ca6cbc1424507668cf54b80325dc01",
"bd0fd461d87fba0d5e61bed6a399acdfc92b12769f9b3178f9752e30f1aeb81d",
"372df01b994c2b14969592fd2e78d27e7ee472a07c7ac3dfdf41d345b2f8e305"
]
forxinrange(1000):
foryinrange(1000):
t="P6"+str(x)+""+str(y)+"255"

#Calculatethehashoftheheader
calculated_hash=calculate_hash(t)


ifcalculated_hashinhash_values:
print("x=",x)
print("y=",y)
print("\n")
```



Codul a generat perechile urmatoare>
C:\Users\Tudor\Desktop\pythonProject\venv\Scripts\python.exe C:\Users\Tudor\Desktop
\pythonProject\main.py
x= 400
y= 433 5

x= 465
y= 464 7

x= 512
y= 512 1

x= 513
y= 613 6

x= 525
y= 489 4

x= 559
y= 530 8

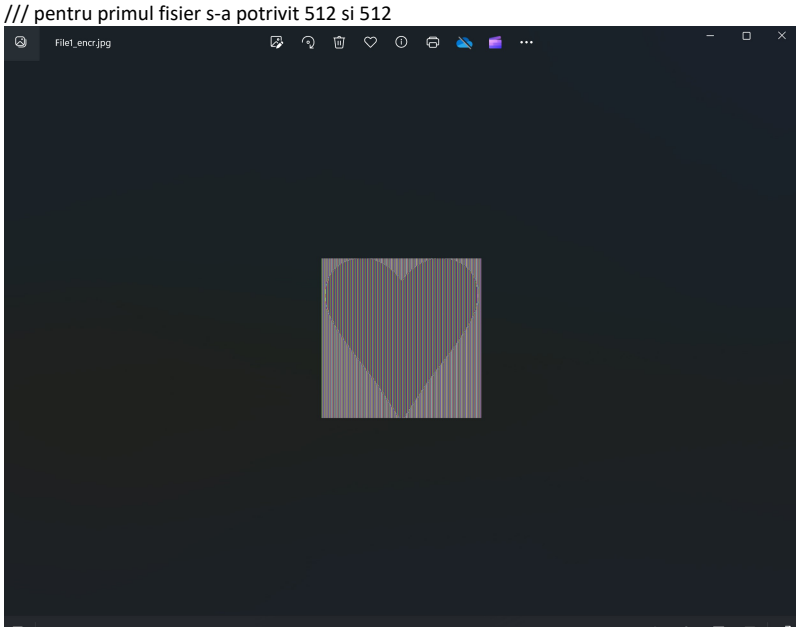
x= 585
y= 577 3

x= 598
y= 605 2

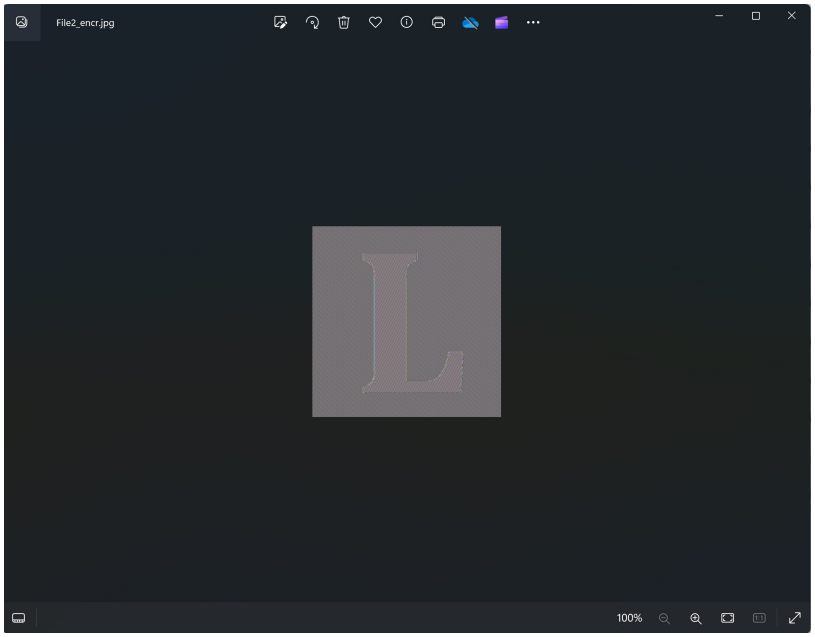
Process finished with exit code 0

Am convertit imaginile cu ajutorul site ului https://convertio.co/
din ppm in jpg

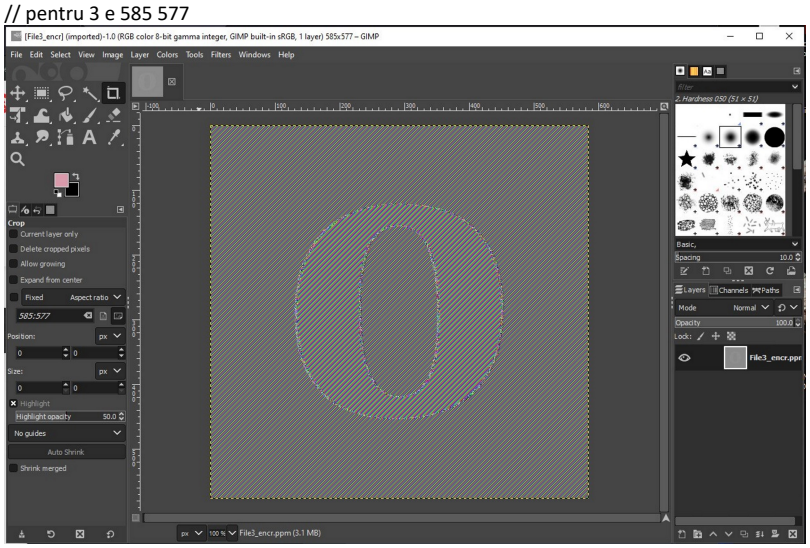
/// pentru primul fisier s-a potrivit 512 si 512



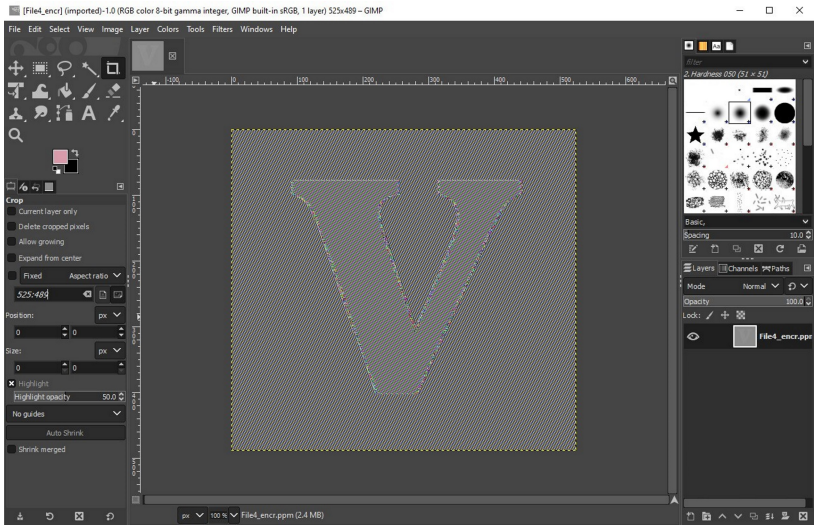
// pentru al doilea am avut 598 605



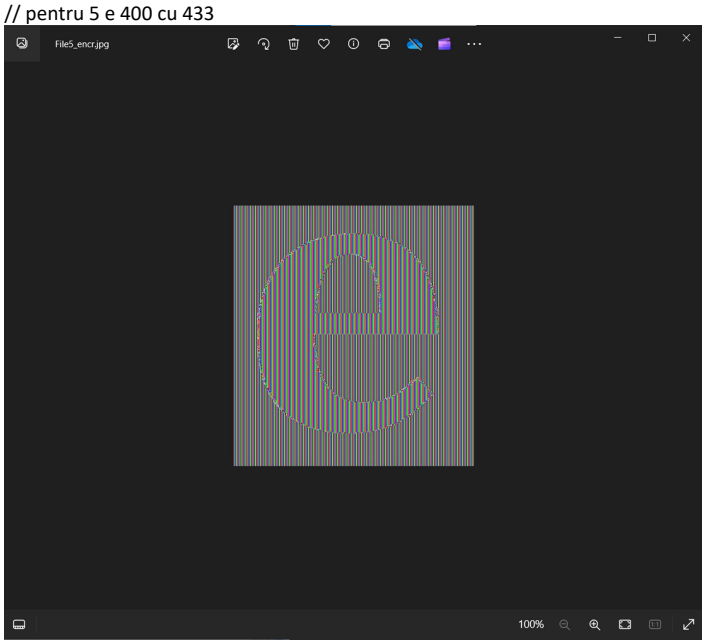
// pentru 3 e 585 577



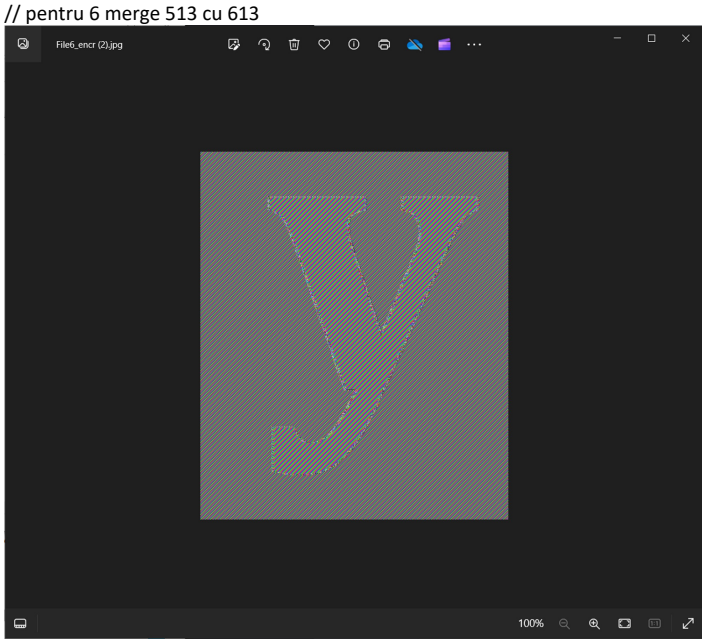
// pentru 4 e 525 489 (aici am deschis cu aplicatia GIMP pentru ca site ul ala e limitat la 10 minute aparent)



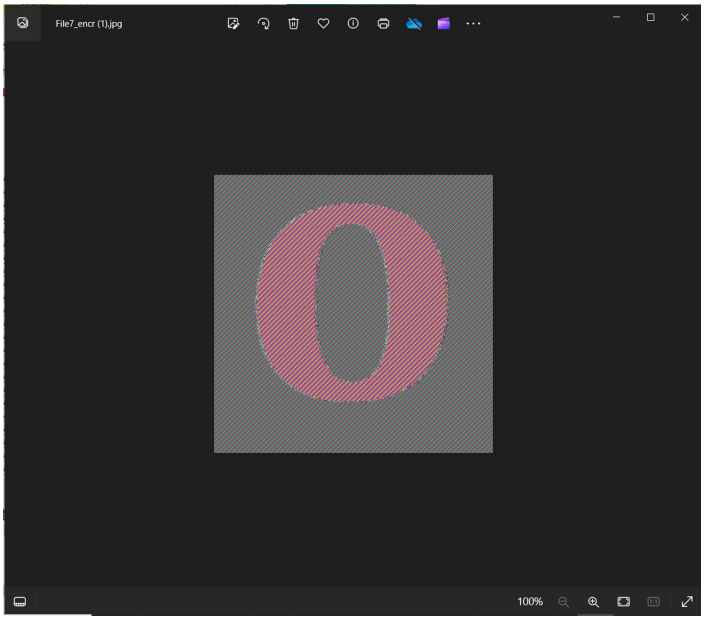
// pentru 5 e 400 cu 433



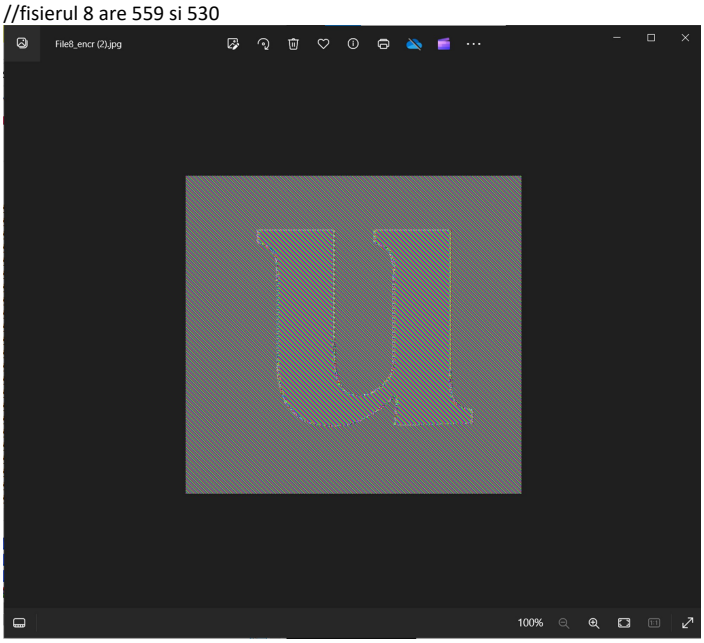
// pentru 6 merge 513 cu 613

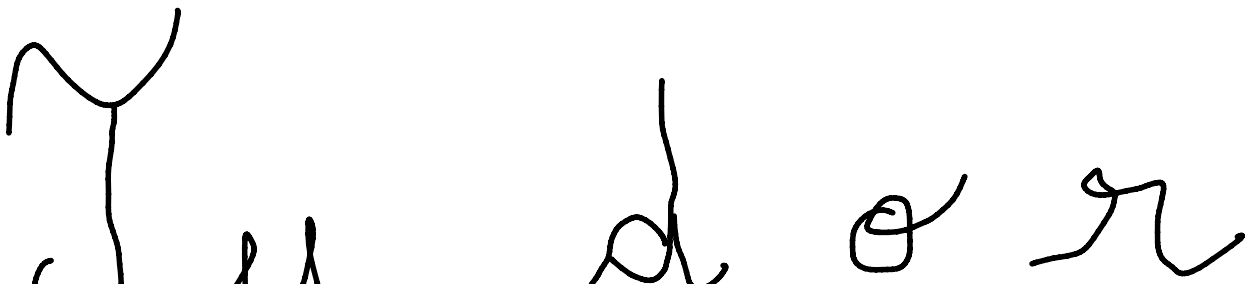


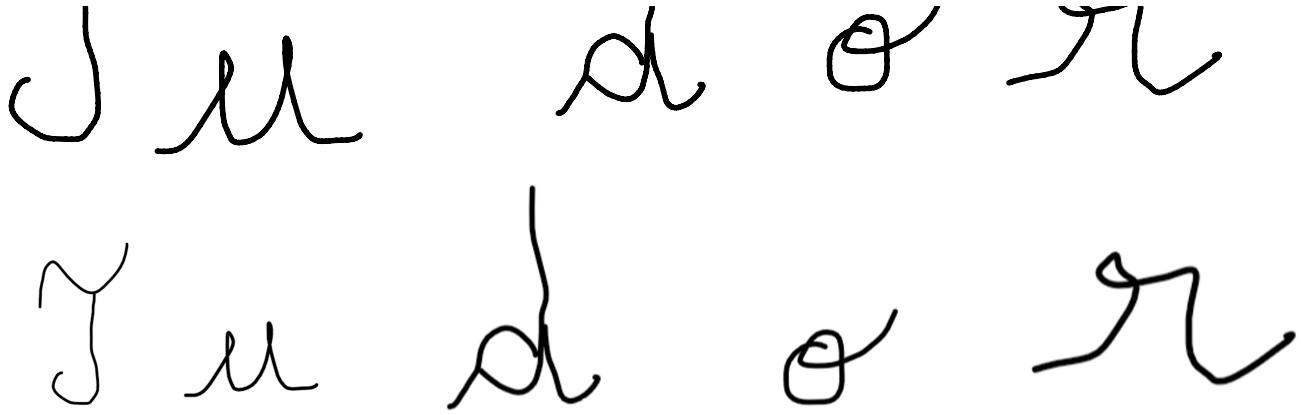
/// fisierul 7 are 465 si 464



//fisierul 8 are 559 si 530







Explicatii pentru partea a doua:

-- deci am luat imaginile astea de mai sus (ss uri) si le am salvat

- le am transformat din png (sau jpg) in ppm cu codul asta (jpgtoppm.py)

```
fromPILimportImage

defpng_to_ppm(input_png,output_ppm):
#ÎncarcăimagineaPNG
img=Image.open(input_png)

#CreeazăunfișierPPMșiscrieheaderulcorespunzător
withopen(output_ppm,'wb')asppm_file:
ppm_file.write(b'P6\n')
ppm_file.write(f"{img.width}{img.height}\n".encode('ascii'))
ppm_file.write(b'255\n')

#Convertireapixelilorșiscriereaînfișier
ppm_file.write(img.tobytes())


input_png='literaT.png'
output_ppm='litT.ppm'
png_to_ppm(input_png,output_ppm)


input_png='literaU.png'
output_ppm='litU.ppm'
png_to_ppm(input_png,output_ppm)


input_png='literaD.png'
output_ppm='litD.ppm'
png_to_ppm(input_png,output_ppm)


input_png='literaO.png'
output_ppm='litO.ppm'
png_to_ppm(input_png,output_ppm)


input_png='literaR.png'
output_ppm='litR.ppm'
png_to_ppm(input_png,output_ppm)
```

-- apoi puteam sa vad headerul pe care l puneam manual intr un fisier txt
-- de acolo le luam si le criptam cu sha-256 cu ajutorul codului asta: (fisierul encryption)

```
importhashlib
```

```
input_file_name='headers'
output_file_name='headers2.txt'

hashed_lines=[]

withopen(input_file_name,'r')asinput_file:
forlineininput_file:
ifline.strip():
hashed_lines.append(hashlib.sha256(line.encode('utf-8')).hexdigest())


withopen(output_file_name,'w')asoutput_file:
forhashed_lineinhashed_lines:
output_file.write(hashed_line)
output_file.write("\n")

print("ok")
```

-- apoi am luat imaginile din ppm si am aplicat peste acestea criptarea de tip ecb (fisierul encryption2)

```
fromcryptography.hazmat.primitives.ciphersimportCipher,algorithms,modes
fromcryptography.hazmat.backendsimportdefault_backend

defencrypt_image_ecb(input_ppm,output_ppm,key):

withopen(input_ppm,'rb')asppm_file:
ppm_content=ppm_file.read()

#Completeazablocurilede16bytes,dacaestenecesar
padding_length=16-(len(ppm_content)%16)
ppm_content+=bytes([padding_length])*padding_length

#CreeazaunobiectCiphercualgoritmulAESinmodulECB
cipher=Cipher(algorithms.AES(key),modes.ECB(),backend=default_backend())

#Cripteazacontinutulimaginii
encryptor=cipher.encryptor()
encrypted_content=encryptor.update(ppm_content)+encryptor.finalize()

#Scriecontinutulcriptatinfisieruldeiesire
withopen(output_ppm,'wb')asoutput_file:
output_file.write(encrypted_content)


input_ppm='litT.ppm'
output_ppm='encryptedT.ppm'


key=b'your_secret_key_'

encrypt_image_ecb(input_ppm,output_ppm,key)
```

asa arata o imagine criptata