**Python day 3**

**CSA0818**

Name :s. sarmash vali

R.no:192372299

**#01**

**Progrm for detecting the Armstrong number**

Code:

a=int(input("enter a number\n"))

n=a

v=0

c=0

while(n!=0):

re=n%10

v=re\*\*3

c=c+v

n=n//10

if c==a:

print(f"{a} is armstrong number\n")

else:

print(f"{a} is not a armstrong number\n")

**output:**

**enter a number**

**153**

**153 is armstrong number**

**#02**

**Progrm for detecting the Happy number**

**Code:**

a=int(input("enter the number\n"))

n=a

su1=0

for i in range (1,n):

while(n>0):

rem=n%10

su=rem\*\*2

su1=su1+su

n=n//10

n=0

n=su1

su1=0

if n==1:

print(f"{ a} is a happy number\n")

else:

print(f"{ a} is a not happy number\n")

**output:**

**enter the number**

**20**

**20 is a not happy number**

**#03**

**Progrm to calculate simple interest**

**Code:**

p=int(input("enter the price money\n"))

t=int(input("enter the time in months\n"))

r=float(input("enter the intrest rate per month\n"))

si=(p\*t\*r)/100

print(f"the simple intrest by the amount\n: { si}")

**output:**

**enter the price money**

**10000**

**enter the time in months**

**4**

**enter the intrest rate per month**

**2.3**

**the simple intrest by the amount**

**: 920.0**

**#04**

**Progrm to print factors of given number**

**Code:**

n=int(input("enter the number to print factores\n"))

count=0

for i in range(1,n+1):

if n%i==0:

count+=1

print(i)

print(f"the number of factores {count}\n")

**output:**

**enter the number to print factores**

**12**

**1**

**2**

**3**

**4**

**6**

**12**

**the number of factores 6**

**#05**

**Progrm to calculate square and cube of the given decimal number**

**Code:**

n=int(input("enter the number to print factores\n"))

sqr=n\*\*2

cube=n\*\*3

print(f"the square {n} is {sqr}\n")

print(f"the cube {n} is {cube}\n")

**output:**

**enter the number to print factores**

**5**

**the square 5 is 25**

**the cube 5 is 125**

**#06**

**Progrm to convert binary to octal and binary to decimal**

**Code:**

b=(input("enter the binary number\n"))

d=int(b,2)

o=oct(d)

print(f"the octal number {b} is {o}\n")

print(f"the decimal number {b} is {d}\n")

**output:**

**enter the binary number**

**1010**

**the octal number 1010 is 0o12**

**the decimal number 1010 is 10**

**#07**

**Progrm to calculate square and cube of the given decimal number**

**Code:**

n1=input("enter the binary value\n")

n2=input("enter the binary value\n")

c=int(n1,2)

d=int(n2,2)

sum=c+d

sum=bin(sum)[2:]

print(sum)

output:

enter the binary value

1

enter the binary value

11

100

**#08**

**Progrm to calculate square and cube of the given decimal number**

Code:

def binary\_to\_decimal(binary):

decimal = 0

for digit in binary:

decimal = decimal\*2 + int(digit)

return decimal

def find\_greatest\_binary\_number(binary1, binary2, binary3):

decimal1 = binary\_to\_decimal(binary1)

decimal2 = binary\_to\_decimal(binary2)

decimal3 = binary\_to\_decimal(binary3)

greatest\_decimal = max(decimal1, decimal2, decimal3)

if greatest\_decimal == decimal1: **output: 110 ,0001 , 1010**

**1010 is the greatest number**

return binary1

elif greatest\_decimal == decimal2:

return binary2

else:

return binary3

binary1 = "1010"

binary2 = "1101"

binary3 = "1001"

greatest\_binary = find\_greatest\_binary\_number(binary1, binary2, binary3)

print(f"The greatest binary number is: {greatest\_binary}")

**#09**

**Progrm to perform matrix multiplication**

Code:

import numpy as np

# Define matrices

matrix1 = np.array([[1, 2], [3, 4]])

matrix2 = np.array([[5, 6], [7, 8]])

# Perform matrix multiplication

result = np.dot(matrix1, matrix2)

print("Matrix 1:")

print(matrix1)

print("\nMatrix 2:")

print(matrix2)

print("\nResult of Matrix Multiplication:")

print(result)

**OUTPUT:**

**Matrix 1:**

**[[1 2]**

**[3 4]]**

**Matrix 2:**

**[[5 6]**

**[7 8]]**

**Result of Matrix Multiplication:**

**[[19 22]**

**[43 50]]**

**#10**

**Progrm to perform matrix multiplication**

Code:

# Matrix addition program

X = [[12,7,3],

[4 ,5,6],

[7 ,8,9]]

Y = [[5,8,1],

[6,7,3],

[4,5,9]]

result = [[0,0,0],

[0,0,0],

[0,0,0]]

for i in range(len(X)):

for j in range(len(X[0])):

result[i][j] = X[i][j] + Y[i][j]

for r in result:

print(r)

**output:**

**[17, 15, 4]**

**[10, 12, 9]**

**[11, 13, 18]**