**CTP Lab Exercise Programs**

**Part-A :: Programs on Control Statements**

**1)Program to determine whether the given number is EVEN or ODD**

n=int(input())

if n%2==0:

print("EVEN")

else:

print("ODD")

**INPUT:**

10

**OUTPUT:**

EVEN

**INPUT:**

345

**OUTPUT:**

ODD

**2) Program to determine whether the given year is leap year or not**

n=int(input())

if (n%4==0 and n%100!=0) or n%400==0:

print("LEAP YEAR")

else:

print("ORDINARY YEAR")

**INPUT:**

2024

**OUTPUT:**

LEAP YEAR

**INPUT:**

2002

**OUTPUT:**

ORDINARY YEAR

**3) Program to determine the greatest of 3 numbers (using elif, nested if else)**

**#USING elif**

a=int(input())

b=int(input())

c=int(input())

if a>=b and a>=c:

print(a)

elif b>=c:

print(b)

else:

print(c)

**#USING nested if else**

a=int(input())

b=int(input())

c=int(input())

if a>=b:

if a>=c:

print(a)

else:

print(c)

else:

if b>=c:

print(b)

else:

print(c)

**INPUT:**

10

20

30

**OUTPUT:**

30

**4) Program to print 3 numbers in descending order**

a=int(input())

b=int(input())

c=int(input())

a1=max(a,b,c)

a3=min(a,b,c)

a2=(a+b+c)-a1-a3

print(a1,a2,a3)

**INPUT:**

22

7909

387

**OUTPUT:**

7909 387 22

**5) Program to display the multiplication table**

n=int(input())

for i in range(1,11):

print(n,'\*',i,'=',n\*i)

**INPUT:**

15

**OUTPUT:**

15 \* 1 = 15

15 \* 2 = 30

15 \* 3 = 45

15 \* 4 = 60

15 \* 5 = 75

15 \* 6 = 90

15 \* 7 = 105

15 \* 8 = 120

15 \* 9 = 135

15 \* 10 = 150

**6) Program to find the sum of first n natural numbers**

n=int(input())

sum=0

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

sum+=i

print(sum)

**INPUT:**

5

**OUTPUT:**

15

**INPUT:**

10

**OUTPUT:**

55

**7) Program to find the sum of squares of first n natural numbers**

n=int(input())

sum=0

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

sum+=i\*\*2

print(sum)

**INPUT:**

10

**OUTPUT:**

385

**8) 12-22+32-42+52-62+-------------+-n2**

n=int(input())

sum=0

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

if i%2==0:

sum-=i\*i

else:

sum+=i\*i

print(sum)

**INPUT:**

5

**OUTPUT:**

15

**9)12-32+52-72+92-112+-------------+-n2**

n=int(input())

sign=1

sum=0

for i in range(1,n+1,2):

sum+=sign\*(i\*i)

sign\*=-1

print(sum)

**INPUT:**

5

**OUTPUT:**  
17

**10)Program to find the sum of digits of the given number**

n=int(input())

sumd=0

while n!=0:

r=n%10

sumd+=r

n=n//10

print(sumd)

**INPUT:**

12345

**OUTPUT:**

15

**11) Program to determine whether the given number is palindrome or not**

n=int(input())

temp=n

sumd=0

rev=0

while n!=0:

r=n%10

rev=r+rev\*10

n=n//10

if rev==temp:

print("Palindrome")

else:

print("Not Palindrome")

**INPUT:**

123321

**OUTPUT:**

Palindrome

**INPUT:**

123

**OUTPUT:**

Not Palindrome

**12) Program to determine whether the given number is Armstrong number or not**

n=int(input())

l=len(str(n))

temp=n

sumd=0

while n!=0:

r=n%10

sumd+=r\*\*l

n=n//10

if sumd==temp:

print("Armstrong")

else:

print("Not Armstrong")

**INPUT:**

153

**OUTPUT:**

Armstrong

**INPUT:**

500

**OUTPUT:**

Not Armstrong

**13) Program to determine whether the given number is PRIME or NOT**

n=int(input())

count=0

for i in range(2,n):

if n%i==0:

count+=1

if count==0:

print("Prime")

else:

print("Not a prime")

**INPUT:**

17

**OUTPUT:**

Prime

**INPUT:**

21

**OUTPUT:**

Not a prime

**14) Programs on Patterns**

**1**

**2 2**

**3 3 3**

**4 4 4 4**

**5 5 5 5 5**

for i in range(1,6):

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

print(i,end=' ')

print()

**OUTPUT:**

1

2 2

3 3 3

4 4 4 4

5 5 5 5 5

**1**

**1 2**

**1 2 3**

**1 2 3 4**

**1 2 3 4 5**

for i in range(1,6):

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

print(j,end=' ')

print()

**OUTPUT:**

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

**\***

**\* \***

**\* \* \***

**\* \* \* \***

**\* \* \* \* \***

for i in range(1,6):

print('\* '\*i)

**OUTPUT:**

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

**1**

**2 1**

**3 2 1**

**4 3 2 1**

**5 4 3 2 1**

for i in range(1,6):

for j in range(i,0,-1):

print(j,end=' ')

print()

**OUTPUT:**

1

2 1

3 2 1

4 3 2 1

5 4 3 2 1

**1 2 3 4 5**

**1 2 3 4**

**1 2 3**

**1 2**

**1**

for i in range(0,6):

for j in range(i+1,6):

print(j,end=' ')

print()

**OUTPUT:**

1 2 3 4 5

2 3 4 5

3 4 5

4 5

5

**5 4 3 2 1**

**4 3 2 1**

**3 2 1**

**2 1**

**1**

for i in range(5,0,-1):

for j in range(i,0,-1):

print(j,end=' ')

print()

**OUTPUT:**

5 4 3 2 1

4 3 2 1

3 2 1

2 1

1

**1**

**2 2**

**3 3 3**

**4 4 4 4**

**5 5 5 5 5**

for i in range(1,6):

for k in range(1,6-i):

print(" ",end="")

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

print(i,end=' ')

print()

**OUTPUT:**

1

2 2

3 3 3

4 4 4 4

5 5 5 5 5

**15) sin x=x-x3/3!+x5/5!-x7/7!+------------**

x=int(input())

k=1

res=0

sign,fact=1,1

while 1:

a=(x\*\*k)

b=fact

res=res+(sign\*(x\*\*k)/fact)

sign=-1\*sign

fact=fact\*(k+1)\*(k+2)

k=k+2

if (a/b)==0:

break

print('sin',x,'=',res)

**INPUT:**

3

**OUTPUT:**

sin 3 = 0.1411200080598671

**15) cos x=1-x2/2!+x4/4!-x6/6!+------------**

x=int(input())

k=2

res=1

sign,fact=-1,2

while 1:

a=(x\*\*k)

b=fact

res=res+(sign\*(x\*\*k)/fact)

sign=-1\*sign

fact=fact\*(k+1)\*(k+2)

k=k+2

if (a/b)==0:

break

print('cos',x,'=',res)

**INPUT:**

3

**OUTPUT:**

**Part-B :: Programs on Functions**

**16) Define recursive and non-recursive functions to find the factorial of given number**

***NON RECURSIVE FUNCTION***

def factorial(n):

fact=1

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

fact=fact\*i

return fact

n=int(input())

print(factorial(n))

**INPUT**:

5

**OUTPUT:**

120

***RECURSIVE FUNCTION***

def factorial(n):

if n==0:

return 1

else:

return n\*factorial(n-1)

n=int(input())

print(factorial(n))

**INPUT:**

5

**OUTPUT:**

120

**17) Define recursive and non-recursive functions to find xn**

***NON RECURSIVE FUNCTION***

def power(x,n):

result=1

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

result\*=x

return result

x=int(input())

n=int(input())

print(power(x,n))

INPUT:

3

4

OUTPUT:

81

***RECURSIVE FUNCTION***

def power(x,n):

if n==0:

return 1

else:

return x\*power(x,n-1)

x=int(input())

n=int(input())

print(power(x,n))

**INPUT:**

3

4

**OUTPUT:**

81

**18) Define recursive and non-recursive functions to find GCD of 2 numbers**

***NON RECURSIVE FUNCTION***

def GCD(a,b):

while b!=0:

temp=a

a=b

b=temp%b

return a

a=int(input())

b=int(input())

if a>b:

print(GCD(a,b))

else:

print(GCD(b,a))

**INPUT:**

5

20

**OUTPUT:**

5

***RECURSIVE FUNCTION***

def GCD(a,b):

if b==0:

return(a)

else:

return GCD(b,a%b)

a=int(input())

b=int(input())

if a>b:

print(GCD(a,b))

else:

print(GCD(b,a))

**INPUT:**

5

20

**OUTPUT:**

5

**19) Define recursive and non-recursive functions to find the nth term in the Fibonacci series**

***NON RECURSIVE FUNCTION***

def fib(n):

a,b=0,1

count=0

while count!=n:

a,b=b,a+b

count+=1

return a

n=int(input())

print(fib(n))

**INPUT:**

6

**OUTPUT:**

8

***RECURSIVE FUNCTION***

def fib(n):

if n == 1 or n == 2:

return 1

else:

return fib(n - 1) + fib(n - 2)

n=int(input())

print(fib(n))

**INPUT:**

6

**OUTPUT:**

8

***NAME:*** BIYYAM AKHILA REDDY

***REGD\_No.:*** 20pa1a0520

***CLASS:*** CSE-A