1. **Python Program for Cycle Sort**

# Python program to implement cycle sort

def cycleSort(array):

writes = 0

# Loop through the array to find cycles to rotate.

for cycleStart in range(0, len(array) - 1):

item = array[cycleStart]

# Find where to put the item.

pos = cycleStart

for i in range(cycleStart + 1, len(array)):

if array[i] < item:

pos += 1

# If the item is already there, this is not a cycle.

if pos == cycleStart:

continue

# Otherwise, put the item there or right after any duplicates.

while item == array[pos]:

pos += 1

array[pos], item = item, array[pos]

writes += 1

# Rotate the rest of the cycle.

while pos != cycleStart:

# Find where to put the item.

pos = cycleStart

for i in range(cycleStart + 1, len(array)):

if array[i] < item:

pos += 1

# Put the item there or right after any duplicates.

while item == array[pos]:

pos += 1

array[pos], item = item, array[pos]

writes += 1

return writes

# driver code

arr = [1, 8, 3, 9, 10, 10, 2, 4 ]

n = len(arr)

cycleSort(arr)

print("After sort : ")

for i in range(0, n) :

print(arr[i], end = \' \')

1. **Python Program for Stooge Sort**

def stoogesort(arr, l, h):

if l >= h:

return

# If first element is smaller

# than last,swap them

if arr[l]>arr[h]:

t = arr[l]

arr[l] = arr[h]

arr[h] = t

# If there are more than 2 elements in

# the array

if h-l+1 > 2:

t = (int)((h-l+1)/3)

# Recursively sort first 2/3 elements

stoogesort(arr, l, (h-t))

# Recursively sort last 2/3 elements

stoogesort(arr, l+t, (h))

# Recursively sort first 2/3 elements

# again to confirm

stoogesort(arr, l, (h-t))

# driver

arr = [2, 4, 5, 3, 1]

n = len(arr)

stoogesort(arr, 0, n-1)

for i in range(0, n):

print(arr[i], end = \' \')

1. **Python Program to print the pattern ‘G’**

# Python program to implement stooge sort

# Python program to print pattern G

def Pattern(line):

pat=""

for i in range(0,line):

for j in range(0,line):

if ((j == 1 and i != 0 and i != line-1) or ((i == 0 or

i == line-1) and j > 1 and j < line-2) or (i == ((line-1)/2)

and j > line-5 and j < line-1) or (j == line-2 and

i != 0 and i != line-1 and i >=((line-1)/2))):

pat=pat+"\*"

else:

pat=pat+" "

pat=pat+"\n"

return pat

# Driver Code

line = 7

print(Pattern(line))

1. **Python Program to print an Inverted Star Pattern**

rows = int(input("Enter the number of rows: "))

for a in range(rows + 1, 0, -1):

for b in range(0, a - 1):

print("\*", end=' ')

print(" ")

1. **Python Program to print double sided stair-case pattern**

# Python3 Program to demonstrate

# staircase pattern

# function definition

def pattern(n):

# for loop for rows

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

# conditional operator

k =i + 1 if(i % 2 != 0) else i

# for loop for printing spaces

for g in range(k,n):

if g>=k:

print(end=" ")

# according to value of k carry

# out further operation

for j in range(0,k):

if j == k - 1:

print(" \* ")

else:

print(" \* ", end = " ")

# Driver code

n = 10

pattern(n)

1. **Python Program to print with your own font**

n = input("Enter the name to print: ")

length = len(n)

l = ""

for a in range(0, length):

b= name[a]

b = b.upper()

if (b == "A"):

print("..######..\n..#....#..\n..######..", end = " ")

print("\n..#....#..\n..#....#..\n")

elif (b == "B"):

print("..######..\n..#....#..\n..#####...", end = " ")

print("\n..#....#..\n..######..\n")

elif (b == "C"):

print("..######..\n..#.......\n..#.......", end = " ")

print("\n..#.......\n..######..\n")

elif (b == "D"):

print("..#####...\n..#....#..\n..#....#..", end = " ")

print("\n..#....#..\n..#####...\n")

elif (b == "E"):

print("..######..\n..#.......\n..#####...", end = " ")

print("\n..#.......\n..######..\n")

elif (b == "F"):

print("..######..\n..#.......\n..#####...", end = " ")

print("\n..#.......\n..#.......\n")

elif (b == "G"):

print("..######..\n..#.......\n..#.####..", end = " ")

print("\n..#....#..\n..#####...\n")

elif (b == "H"):

print("..#....#..\n..#....#..\n..######..", end = " ")

print("\n..#....#..\n..#....#..\n")

elif (b == "I"):

print("..######..\n....##....\n....##....", end = " ")

print("\n....##....\n..######..\n")

elif (b == "J"):

print("..######..\n....##....\n....##....", end = " ")

print("\n..#.##....\n..####....\n")

elif (b == "K"):

print("..#...#...\n..#..#....\n..##......", end = " ")

print("\n..#..#....\n..#...#...\n")

elif (b == "L"):

print("..#.......\n..#.......\n..#.......", end = " ")

print("\n..#.......\n..######..\n")

elif (b == "M"):

print("..#....#..\n..##..##..\n..#.##.#..", end = " ")

print("\n..#....#..\n..#....#..\n")

elif (b == "N"):

print("..#....#..\n..##...#..\n..#.#..#..", end = " ")

print("\n..#..#.#..\n..#...##..\n")

elif (b == "O"):

print("..######..\n..#....#..\n..#....#..", end = " ")

print("\n..#....#..\n..######..\n")

elif (b == "P"):

print("..######..\n..#....#..\n..######..", end = " ")

print("\n..#.......\n..#.......\n")

elif (b == "Q"):

print("..######..\n..#....#..\n..#.#..#..", end = " ")

print("\n..#..#.#..\n..######..\n")

elif (b == "R"):

print("..######..\n..#....#..\n..#.##...", end = " ")

print("\n..#...#...\n..#....#..\n")

elif (b == "S"):

print("..######..\n..#.......\n..######..", end = " ")

print("\n.......#..\n..######..\n")

elif (b == "T"):

print("..######..\n....##....\n....##....", end = " ")

print("\n....##....\n....##....\n")

elif (b == "U"):

print("..#....#..\n..#....#..\n..#....#..", end = " ")

print("\n..#....#..\n..######..\n")

elif (b == "V"):

print("..#....#..\n..#....#..\n..#....#..", end = " ")

print("\n...#..#...\n....##....\n")

elif (b == "W"):

print("..#....#..\n..#....#..\n..#.##.#..", end = " ")

print("\n..##..##..\n..#....#..\n")

elif (b == "X"):

print("..#....#..\n...#..#...\n....##....", end = " ")

print("\n...#..#...\n..#....#..\n")

elif (b == "Y"):

print("..#....#..\n...#..#...\n....##....", end = " ")

print("\n....##....\n....##....\n")

elif (b == "Z"):

print("..######..\n......#...\n.....#....", end = " ")

print("\n....#.....\n..######..\n")

elif (b == " "):

print("..........\n..........\n..........", end = " ")

print("\n..........\n\n")

elif (b == "."):

print("----..----\n\n")

1. **Get Current Date and Time using Python**

from datetime import date

today = date.today()

print("Today's date:", today)

1. **Python | Find yesterday’s, today’s and tomorrow’s date**

from datetime import datetime, timedelta

# Get today's date

presentday = datetime.now() # or presentday = datetime.today()

# Get Yesterday

yesterday = presentday - timedelta(1)

# Get Tomorrow

tomorrow = presentday + timedelta(1)

# strftime() is to format date according to

# the need by converting them to string

print("Yesterday = ", yesterday.strftime('%d-%m-%Y'))

print("Today = ", presentday.strftime('%d-%m-%Y'))

print("Tomorrow = ", tomorrow.strftime('%d-%m-%Y'))

1. **Python 24 hour program to convert time from 12 hour to format**

# Python program to find yesterday,

# today and tomorrow

# Import datetime and timedelta

# class from datetime module

from datetime import datetime, timedelta

# Get today's date

presentday = datetime.now() # or presentday = datetime.today()

# Get Yesterday

yesterday = presentday - timedelta(1)

# Get Tomorrow

tomorrow = presentday + timedelta(1)

# strftime() is to format date according to

# the need by converting them to string

print("Yesterday = ", yesterday.strftime('%d-%m-%Y'))

print("Today = ", presentday.strftime('%d-%m-%Y'))

print("Tomorrow = ", tomorrow.strftime('%d-%m-%Y'))