Python Programming

Data Handling:

Question 1: Write a program to obtain principal amount, rate of interest and time from user and compute simple interest.

Solution

```
p = float(input("Enter principal: "))
r = float(input("Enter rate: "))
t = float(input("Enter time: "))

si = p * r * t / 100

print("Simple Interest =", si)
```

Output

Enter principal: 55000.75
Enter rate: 14.5
Enter time: 3
Simple Interest = 23925.32625

Question 2: Write a program to obtain temperatures of 7 days (Monday, Tuesday ... Sunday) and then display average temperature of the week.

Solution

```
d1 = float(input("Enter Sunday Temperature: "))
d2 = float(input("Enter Monday Temperature: "))
d3 = float(input("Enter Tuesday Temperature: "))
d4 = float(input("Enter Wednesday Temperature: "))
d5 = float(input("Enter Thursday Temperature: "))
d6 = float(input("Enter Friday Temperature: "))
d7 = float(input("Enter Saturday Temperature: "))

avg = (d1 + d2 + d3 + d4 + d5 + d6 + d7) / 7

print("Average Temperature =", avg)
```

Output

Enter Sunday Temperature: 21.6
Enter Monday Temperature: 22.3
Enter Tuesday Temperature: 24.5
Enter Wednesday Temperature: 23.0
Enter Thursday Temperature: 23.7
Enter Friday Temperature: 24.2
Enter Saturday Temperature: 25
Average Temperature = 23.47142857142857

Question 3: Write a program to obtain x, y, z from user and calculate expression : $4x^4 + 3y^3 + 9z + 6\pi$

Solution

```
import math
x = int(input("Enter x: "))
y = int(input("Enter y: "))
z = int(input("Enter z: "))
res = 4 * x ** 4 + 3 * y ** 3 + 9 * z + 6 * math.pi
print("Result =", res)
```

Output

Enter x: 2
Enter y: 3
Enter z: 5
Result = 208.84955592153875

Question 4: Write a program that reads a number of seconds and prints it in form : mins and seconds, e.g., 200 seconds are printed as 3 mins and 20 seconds.
[Hint. use // and % to get minutes and seconds]

Solution

```
totalSecs = int(input("Enter seconds: "))
mins = totalSecs // 60
secs = totalSecs % 60
print(mins, "minutes and", secs, "seconds")
```
Output
Enter seconds: 200
3 minutes and 20 seconds

Question 5: Write a program to take year as input and check if it is a leap year or not.
Solution
```
y = int(input("Enter year to check: "))
print(y % 4 and "Not a Leap Year" or "Leap Year")
```
Output
Enter year to check: 2020
Leap Year

Question 6: Write a program to take two numbers and print if the first number is fully divisible by second number or not.
Solution
```
x = int(input("Enter first number: "))
y = int(input("Enter second number: "))
print(x % y and "Not Fully Divisible" or "Fully Divisible")
```
Output
Enter first number: 4
Enter second number: 2
Fully Divisible

Question 7: Write a program to take a 2-digit number and then print the reversed number. That is, if the input given is 25, the program should print 52.
Solution
```
x = int(input("Enter a two digit number: "))
y = x % 10 * 10 + x // 10
print("Reversed Number:", y)
```
Output
Enter a two digit number: 25
Reversed Number: 52

Question 8: Try writing program (similar to previous one) for three digit number i.e., if you input 123, the program should print 321.
Solution
```
x = int(input("Enter a three digit number: "))
d1 = x % 10
x //= 10
d2 = x % 10
x //= 10
d3 = x % 10
y = d1 * 100 + d2 * 10 + d3
print("Reversed Number:", y)
```
Output
Enter a three digit number: 123
Reversed Number: 321

Question 9: Write a program to take two inputs for day, month and then calculate which day of the year, the given date is. For simplicity, take 30 days for all months. For example, if you give input as: Day3, Month2 then it should print "Day of the year : 33".
Solution
```
d = int(input("Enter day: "))
m = int(input("Enter month: "))

n = (m - 1) * 30 + d
```

print("Day of the year:", n)

Output

Enter day: 3
Enter month: 2
Day of the year: 33

Question 10: Write a program that asks a user for a number of years, and then prints out the number of days, hours, minutes, and seconds in that number of years.

How many years? 10
10.0 years is:
3650.0 days
87600.0 hours
5256000.0 minutes
315360000.0 seconds

Solution

```
y = float(input("How many years? "))
d = y * 365
h = d * 24
m = h * 60
s = m * 60

print(y, "years is:")
print(d, "days")
print(h, "hours")
print(m, "minutes")
print(s, "seconds")
```

Output

How many years? 10
10.0 years is:
3650.0 days
87600.0 hours
5256000.0 minutes
315360000.0 seconds

Question 11: Write a program that inputs an age and print age after 10 years as shown below:

What is your age? 17
In ten years, you will be 27 years old!

Solution

```
a = int(input("What is your age? "))
print("In ten years, you will be", a + 10, "years old!")
```

Output

What is your age? 17
In ten years, you will be 27 years old!

Question 12: Write a program whose three sample runs are shown below:

Sample Run 1:
Random number between 0 and 5 (A) : 2
Random number between 0 and 5 (B) :5.
A to the power B = 32
Sample Run 2:
Random number between 0 and 5 (A) : 4
Random number between 0 and 5 (B) :3.
A to the power B = 64
Sample Run 3:
Random number between 0 and 5 (A) : 1
Random number between 0 and 5 (B) :1.
A to the power B = 1

Solution

```
import random
```

```
a = random.randint(0, 5)
b = random.randint(0, 5)
c = a ** b

print("Random number between 0 and 5 (A) :", a)
print("Random number between 0 and 5 (B) :", b)
print("A to the power B =", c)
```

Output

Random number between 0 and 5 (A) : 5
Random number between 0 and 5 (B) : 3
A to the power B = 125

Question 13: Write a program that generates six random numbers in a sequence created with (start, stop, step). Then print the mean, median and mode of the generated numbers.

Solution

```
import random
import statistics

start = int(input("Enter start: "))
stop = int(input("Enter stop: "))
step = int(input("Enter step: "))

a = random.randrange(start, stop, step)
b = random.randrange(start, stop, step)
c = random.randrange(start, stop, step)
d = random.randrange(start, stop, step)
e = random.randrange(start, stop, step)
f = random.randrange(start, stop, step)

print("Generated Numbers:")
print(a, b, c, d, e, f)

seq = (a, b, c, d, e, f)

mean = statistics.mean(seq)
median = statistics.median(seq)
mode = statistics.mode(seq)

print("Mean =", mean)
print("Median =", median)
print("Mode =", mode)
```

Output

Enter start: 100
Enter stop: 500
Enter step: 5
Generated Numbers:
235 255 320 475 170 325
Mean = 296.6666666666667
Median = 287.5
Mode = 235

Question 14: Write a program to generate 3 random integers between 100 and 999 which is divisible by 5.

Solution

```
import random

a = random.randrange(100, 999, 5)
b = random.randrange(100, 999, 5)
c = random.randrange(100, 999, 5)
```

```
print("Generated Numbers:", a, b, c)
```
Output
Generated Numbers: 885 825 355

Question 15: Write a program to generate 6 digit random secure OTP between 100000 to 999999.
Solution
```
import random

otp = random.randint(100000, 999999);

print("OTP:", otp);
```
Output
OTP: 553072

Question 16: Write a program to generate 6 random numbers and then print their mean, median and mode.
Solution
```
import random
import statistics

a = random.random()
b = random.random()
c = random.random()
d = random.random()
e = random.random()
f = random.random()

print("Generated Numbers:")
print(a, b, c, d, e, f)

seq = (a, b, c, d, e, f)

mean = statistics.mean(seq)
median = statistics.median(seq)
mode = statistics.mode(seq)

print("Mean =", mean)
print("Median =", median)
print("Mode =", mode)
```
Output
Generated Numbers:
0.47950245404109626 0.6908539320958872 0.12445888663826654 0.13613724999684718 0.37709141355821396 0.6369609321575742
Mean = 0.40750081141464756
Median = 0.4282969337996551
Mode = 0.47950245404109626

Question 17: Write a program to find a side of a right angled triangle whose two sides and an angle is given.
Solution
```
import math

a = float(input("Enter base: "))
b = float(input("Enter height: "))
x = float(input("Enter angle: "))

c = math.sqrt(a ** 2 + b ** 2)

print("Hypotenuse =", c)
```

Output
Enter base: 10.5
Enter height: 5.5
Enter angle: 60
Hypotenuse = 11.853269591129697

Question 18: Write a program to calculate the radius of a sphere whose area ($4\pi r^2$) is given.
Solution
```
import math

area = float(input("Enter area of sphere: "))

r = math.sqrt(area / (4 * math.pi))

print("Radius of sphere =", r)
```
Output
Enter area of sphere: 380.14
Radius of sphere = 5.50005273006328

Question 19: Write a program that inputs a string and then prints it equal to number of times its length, e.g.,
Enter string : "eka"
Result ekaekaeka
Solution
```
str = input("Enter string: ")
len = len(str)
opStr = str * len
print("Result", opStr)
```
Output
Enter string: eka
Result ekaekaeka

Question 20: Find the volume of the cylinder ($\pi r^2 h$) as shown:
Radius = 8 cm
Height = 15 cm
Solution
```
import math

r = 8
h = 15
v = math.pi * r * r * h

print("Volume of Cylinder =", v)
```
Output
Volume of Cylinder = 3015.928947446201

Question 21: Write a program to calculate the area of an equilateral triangle. (area = ($\sqrt{3}$ / 4) * side * side).
Solution
```
import math

side = float(input("Enter side: "))
area = math.sqrt(3) / 4 * side * side

print("Area of triangle =", area)
```
Output
Enter side: 5
Area of triangle = 10.825317547305481

Question 22: Write a program to input the radius of a sphere and calculate its volume ($V = 4/3\pi r^3$)

Solution

```
import math

r = float(input("Enter radius of sphere: "))
v = 4 / 3 * math.pi * r ** 3

print("Volume of sphere = ", v)
```

Output

Enter radius of sphere: 3.5
Volume of sphere = 179.59438003021648

Question 23: Write a program to calculate amount payable after simple interest.

Solution

```
p = float(input("Enter principal: "))
r = float(input("Enter rate: "))
t = float(input("Enter time: "))

si = p * r * t / 100
amt = p + si

print("Amount Payable =", amt)
```

Output

Enter principal: 55000.75
Enter rate: 14.5
Enter time: 3
Amount Payable = 78926.07625

Question 24: Write a program to calculate amount payable after compound interest.

Solution

```
p = float(input("Enter principal: "))
r = float(input("Enter rate: "))
t = float(input("Enter time: "))

amt = p * (1 + r / 100) ** t

print("Amount Payable =", amt)
```

Output

Enter principal: 15217.75
Enter rate: 9.2
Enter time: 3
Amount Payable = 19816.107987312007

Question 25: Write a program to compute $(a + b)^3$ using the formula $a^3 + b^3 + 3a^2b + 3ab^2$

Solution

```
a = int(input("Enter a: "))
b = int(input("Enter b: "))
res = a ** 3 + b ** 3 + 3 * a ** 2 * b + 3 * a * b ** 2

print("Result =", res)
```

Output

Enter a: 3
Enter b: 5
Result = 512

Flow of Controls

1. Write a Python script that asks the user to enter a length in centimetres. If the user enters a negative length, the program should tell the user that the entry is invalid. Otherwise, the program should convert the length to inches and print out the result. There are 2.54 centimetres in an inch.

Solution

```
len = int(input("Enter length in cm: "))
if len < 0:
    print("Invalid input")
else:
    inch = len / 2.54
    print(len, "centimetres is equal to", inch, "inches")
```

Output

```
Enter length in cm: 150
150 centimetres is equal to 59.05511811023622 inches
```

2. A store charges ₹120 per item if you buy less than 10 items. If you buy between 10 and 99 items, the cost is ₹100 per item. If you buy 100 or more items, the cost is ₹70 per item. Write a program that asks the user how many items they are buying and prints the total cost.

Solution

```
n = int(input("Enter number of items: "))

cost = 0

if n >= 100 :
    cost = n * 70
elif n >= 10 :
    cost = n * 100
else :
    cost = n * 120

print("Total Cost =", cost)
```

Output

```
Enter number of items: 58
Total Cost = 5800
```

3. Write a program that reads from user — (i) an hour between 1 to 12 and (ii) number of hours ahead. The program should then print the time after those many hours, e.g.,

Enter hour between 1-12 : 9
How many hours ahead : 4
Time at that time would be : 1 O'clock

Solution

```
hr = int(input("Enter hour between 1-12 : "))
n = int(input("How many hours ahead : "))

s = hr + n

if s > 12:
    s -= 12

print("Time at that time would be : ", s, "O'clock")
```

Output

```
Enter hour between 1-12 : 9
How many hours ahead : 4
Time at that time would be :  1 O'clock
```

4. Write a program that asks the user for two numbers and prints Close if the numbers are within .001 of each other and Not close otherwise.

Solution

```
a = float(input("Enter first number: "))
b = float(input("Enter second number: "))

d = 0
if a > b :
    d = a - b
else :
    d = b - a

if d <= 0.001 :
    print("Close")
else :
    print("Not Close")
```

Output

```
Enter first number: 10.12345
Enter second number: 10.12354
Close
```

5. A year is a leap year if it is divisible by 4, except that years divisible by 100 are not leap years unless they are also divisible by 400. Write a program that asks the user for a year and prints out whether it is a leap year or not.

Solution

```
year = int(input("Enter year: "))

if year % 400 == 0 :
    print(year, "is a Leap Year")
elif year % 100 == 0 :
    print(year, "is not a Leap Year")
elif year % 4 == 0 :
    print(year, "is a Leap Year")
else :
    print(year, "is not a Leap Year")
```

Output

```
Enter year: 1800
1800 is not a Leap Year
```

6. Write a program to input length of three sides of a triangle. Then check if these sides will form a triangle or not.
(Rule is: a+b>c;b+c>a;c+a>b)

Solution

```
a = int(input("Enter first side : "))
b = int(input("Enter second side : "))
c = int(input("Enter third side : "))

if a + b > c and b + c > a and a + c > b :
    print("Triangle Possible")
else :
    print("Triangle Not Possible")
```

Output

```
Enter first side : 3
Enter second side : 5
```

```
Enter third side : 6
Triangle Possible
```

7. Write a short program to input a digit and print it in words.

Solution

```
d = int(input("Enter a digit(0-9): "))

if d == 0 :
    print("Zero")
elif d == 1 :
    print("One")
elif d == 2 :
    print("Two")
elif d == 3 :
    print("Three")
elif d == 4 :
    print("Four")
elif d == 5 :
    print("Five")
elif d == 6 :
    print("Six")
elif d == 7 :
    print("Seven")
elif d == 8 :
    print("Eight")
elif d == 9 :
    print("Nine")
else :
    print("Invalid Digit")
```

Output

```
Enter a digit(0-9): 6
Six
```

8. Write a short program to check whether square root of a number is prime or not.

Solution

```
import math

n = int(input("Enter a number: "))
sr = math.sqrt(n)
c = 0

for i in range(1, int(sr + 1)) :
    if (sr % i == 0) :
        c += 1

if c == 2 :
    print("Square root is prime")
else :
    print("Square root is not prime")
```

Output

```
Enter a number: 49
Square root is prime
```

9. Write a short program to print first n odd numbers in descending order.

Solution

```
n = int(input("Enter n: "))
x = n * 2 - 1

for i in range(x, 0, -2) :
    print(i)
```

Output

```
Enter n: 5
9
7
5
3
1
```

10. Write a short program to print the following series :

(i) 1 4 7 10 ..... 40.

(ii) 1 -4 7 -10 ..... -40

Solution

```
print("First Series:")
for i in range(1, 41, 3) :
    print(i, end = ' ')

print("\nSecond Series:")
x = 1
for i in range(1, 41, 3) :
    print(i * x, end = ' ')
    x *= -1
```

Output

```
First Series:
1 4 7 10 13 16 19 22 25 28 31 34 37 40
Second Series:
1 -4 7 -10 13 -16 19 -22 25 -28 31 -34 37 -40
```

11. Write a short program to find average of list of numbers entered through keyboard.

Solution

```
sum = count = 0

print("Enter numbers")
print("(Enter 'q' to see the average)")

while True :
    n = input()
    if n == 'q' or n == 'Q' :
        break
    else :
        sum += int(n)
        count += 1

avg = sum / count
print("Average = ", avg)
```

Output

```
Enter numbers
(Enter 'q' to see the average)
```

```
2
5
7
15
12
q
Average =  8.2
```

12. Write a program to input 3 sides of a triangle and print whether it is an equilateral, scalene or isosceles triangle.

Solution

```
a = int(input("Enter first side : "))
b = int(input("Enter second side : "))
c = int(input("Enter third side : "))

if a == b and b == c :
    print("Equilateral Triangle")
elif a == b or b == c or c == a:
    print("Isosceles Triangle")
else :
    print("Scalene Triangle")
```

Output

```
Enter first side : 10
Enter second side : 5
Enter third side : 10
Isosceles Triangle
```

13. Write a program to take an integer a as an input and check whether it ends with 4 or 8. If it ends with 4, print "ends with 4", if it ends with 8, print "ends with 8", otherwise print "ends with neither".

Solution

```
a = int(input("Enter an integer: "))

if a % 10 == 4 :
    print("ends with 4")
elif a % 10 == 8 :
    print("ends with 8")
else :
    print("ends with neither")
```

Output

```
Enter an integer: 18
ends with 8
```

14. Write a program to take N (N > 20) as an input from the user. Print numbers from 11 to N. When the number is a multiple of 3, print "Tipsy", when it is a multiple of 7, print "Topsy". When it is a multiple of both, print "TipsyTopsy".

Solution

```
n = int(input("Enter a number greater than 20: "))
if n <= 20 :
    print("Invalid Input")
else :
    for i in range(11, n + 1) :
        print(i)
        if i % 3 == 0 and i % 7 == 0 :
```

```
            print("TipsyTopsy")
        elif i % 3 == 0 :
            print("Tipsy")
        elif i % 7 == 0 :
            print("Topsy")
```

Output

```
Enter a number greater than 20: 25
11
12
Tipsy
13
14
Topsy
15
Tipsy
16
17
18
Tipsy
19
20
21
TipsyTopsy
22
23
24
Tipsy
25
```

15. Write a short program to find largest number of a list of numbers entered through keyboard.

Solution

```
print("Enter numbers:")
print("(Enter 'q' to see the result)")

l = input()

if l != 'q' and l != 'Q' :
    l = int(l)
    while True:
        n = input()
        if n == 'q' or n == 'Q' :
            break
        n = int(n)
        if n > l :
            l = n
    print("Largest Number =", l)
```

Output

```
Enter numbers:
(Enter 'q' to see the result)
3
5
8
```

```
2
4
q
Largest Number = 8
```

16. Write a program to input N numbers and then print the second largest number.

Solution

```
n = int(input("How many numbers you want to enter? "))
if n > 1 :
    l = int(input())   # Assume first input is largest
    sl = int(input())  # Assume second input is second largest
    if sl > l :
        t = sl
        sl = l
        l = t
    for i in range(n - 2) :
        a = int(input())
        if a > l :
            sl = l
            l = a
        elif a > sl :
            sl = a
    print("Second Largest Number =", sl)
else :
    print("Please enter more than 1 number")
```

Output

```
How many numbers you want to enter? 5
55
25
36
12
18
Second Largest Number = 36
```

17. Given a list of integers, write a program to find those which are palindromes. For example, the number 4321234 is a palindrome as it reads the same from left to right and from right to left.

Solution

```
print("Enter numbers:")
print("(Enter 'q' to stop)")

while True :
    n = input()
    if n == 'q' or n == 'Q' :
        break
    n = int(n)
    t = n
    r = 0
    while (t != 0) :
        d = t % 10
        r = r * 10 + d
        t = t // 10
    if (n == r) :
```

```
        print(n, "is a Palindrome Number")
    else :
        print(n, "is not a Palindrome Number")
```

Output

```
Enter numbers:
(Enter 'q' to stop)
67826
67826 is not a Palindrome Number
4321234
4321234 is a Palindrome Number
256894
256894 is not a Palindrome Number
122221
122221 is a Palindrome Number
q
```

18. Write a complete Python program to do the following :
(i) read an integer X.
(ii) determine the number of digits n in X.
(iii) form an integer Y that has the number of digits n at ten's place and the most significant digit of X at one's place.
(iv) Output Y.
(For example, if X is equal to 2134, then Y should be 42 as there are 4 digits and the most significant number is 2).

Solution

```
x = int(input("Enter an integer: "))
temp = x
count = 0
digit = -1

while temp != 0 :
    digit = temp % 10
    count += 1
    temp = temp // 10

y = count * 10 + digit

print("Y =", y)
```

Output

```
Enter an integer: 2134
Y = 42
```

19. Write a Python program to print every integer between 1 and n divisible by m. Also report whether the number that is divisible by m is even or odd.

Solution

```
m = int(input("Enter m: "))
n = int(input("Enter n: "))
for i in range(1, n) :
    if i % m == 0 :
        print(i, "is divisible by", m)
        if i % 2 == 0 :
            print(i, "is even")
        else :
            print(i, "is odd")
```

Output

```
Enter m: 3
Enter n: 20
3 is divisible by 3
3 is odd
6 is divisible by 3
6 is even
9 is divisible by 3
9 is odd
12 is divisible by 3
12 is even
15 is divisible by 3
15 is odd
18 is divisible by 3
18 is even
```

20a. Write Python programs to sum the given sequences:
2/9 - 5/13 + 8/17 ..... (print 7 terms)

Solution

```
n = 2 #numerator initial value
d = 9 #denominator initial value
m = 1 #to add/subtract alternate terms
sum = 0

for i in range(7) :
    t = n / d
    sum += t * m
    n += 3
    d += 4
    m *= -1

print("Sum =", sum)
```

Output

```
Sum = 0.3642392586003134
```

20b. Write Python programs to sum the given sequences:
$1^2 + 3^2 + 5^2 + ... + n^2$ (Input n)

Solution

```
n = int(input("Enter the value of n: "))

i = 1
sum = 0

while i <= n :
    sum += i ** 2
    i += 2

print("Sum =", sum)
```

Output

```
Enter the value of n: 9
Sum = 165
```

21. Write a Python program to sum the sequence:
1 + 1/1! + 1/2! + 1/3! + ... + 1/n! (Input n)

```
Solution
n = int(input("Enter the value of n: "))
sum = 0

for i in range(n + 1) :
    fact = 1
    for j in range(1, i) :
        fact *= j
    term = 1 / fact
    sum += term

print("Sum =", sum)
```

Output

```
Sum = 3.708333333333333
```

22. Write a program to accept the age of n employees and count the number of persons in the following age group:

(i) 26 - 35
(ii) 36 - 45
(iii) 46 - 55

Solution

```
n = int(input("Enter the value of n: "))
g1 = g2 = g3 = 0

for i in range(1, n + 1) :
    age = int(input("Enter employee age: "))
    #We have used chained comparison operators
    if 26 <= age <= 35 :
        g1 += 1
    elif 36 <= age <= 45 :
        g2 += 1
    elif 46 <= age <= 55 :
        g3 += 1

print("Employees in age group 26 - 35: ", g1)
print("Employees in age group 36 - 45: ", g2)
print("Employees in age group 46 - 55: ", g3)
```

Output

```
Enter the value of n: 10
Enter employee age: 45
Enter employee age: 53
Enter employee age: 28
Enter employee age: 32
Enter employee age: 34
Enter employee age: 49
Enter employee age: 30
Enter employee age: 38
Enter employee age: 33
Enter employee age: 53
Employees in age group 26 - 35:  5
Employees in age group 36 - 45:  2
Employees in age group 46 - 55:  3
```

23a. Write programs to find the sum of the following series:
$x - x^2/2! + x^3/3! - x^4/4! + x^5/5! - x^6/6!$ (Input x)

Solution

```
x = int(input("Enter the value of x: "))

sum = 0
m = 1

for i in range(1, 7) :
    fact = 1
    for j in range(1, i+1) :
        fact *= j
    term = x ** i / fact
    sum += term * m
    m = m * -1

print("Sum =", sum)
```

Output

```
Enter the value of x: 2
Sum = 0.8444444444444444
```

23b. Write programs to find the sum of the following series:
$x + x^2/2 + x^3/3 + ... + x^n/n$ (Input x and n both)

Solution

```
x = int(input("Enter the value of x: "))
n = int(input("Enter the value of n: "))

sum = 0

for i in range(1, n + 1) :
    term = x ** i / i
    sum += term

print("Sum =", sum)
```

Output

```
Enter the value of x: 2
Enter the value of n: 5
Sum = 17.066666666666666
```

24a. Write programs to print the following shapes:

```
  *
 * *
* * *
 * *
  *
```

Solution

```
n = 3 # number of rows

# upper half
for i in range(n) :
    for j in range(n, i+1, -1) :
        print(' ', end = '')
    for k in range(i+1) :
        print('*', end = ' ')
```

```
    print()

# lower half
for i in range(n-1) :
    for j in range(i + 1) :
        print(' ', end = '')
    for k in range(n-1, i, -1) :
        print('*', end = ' ')
    print()
```

Output

```
  *
 * *
* * *
 * *
  *
```

24b. Write programs to print the following shapes:

```
*
* *
* * *
* *
*
```

Solution

```
n = 3 # number of rows

# upper half
for i in range(n) :
    for k in range(i+1) :
        print('*', end = ' ')
    print()

# lower half
for i in range(n-1) :
    for k in range(n-1, i, -1) :
        print('*', end = ' ')
    print()
```

Output

```
*
* *
* * *
* *
*
```

24c. Write programs to print the following shapes:

```
  *
 * *
*   *
 * *
  *
```

Solution

```
n = 3 # number of rows

# upper half
for i in range(1, n+1) :
    # for loop for initial spaces
```

```
    for j in range(n, i, -1) :
        print(' ', end = '')

    #while loop for * and spaces
    x = 1
    while x < 2 * i :
        if x == 1 or x == 2 * i - 1 :
            print('*', end = '')
        else :
            print(' ', end = '')
        x += 1
    print()

# lower half
for i in range(n-1, 0, -1) :
    # for loop for initial spaces
    for j in range(n, i, -1) :
        print(' ', end = '')

    #while loop for * and spaces
    x = 1
    while x < 2 * i :
        if x == 1 or x == 2 * i - 1 :
            print('*', end = '')
        else :
            print(' ', end = '')
        x += 1
    print()
```

Output

```
 *
 **
* *
 **
 *
```

24d. Write programs to print the following shapes:

```
*
**
* *
*  *
* *
**
*
```

Solution

```
n = 4 # number of row

#upper half
for i in range(1, n+1) :
    #while loop for * and spaces
    x = 1
    while x < 2 * i :
        if x == 1 or x == 2 * i - 1 :
            print('*', end = '')
```

```
        else :
            print(' ', end = '')
        x += 1
    print()

#lower half
for i in range(n-1, 0, -1) :
    #while loop for * and spaces
    x = 1
    while x < 2 * i :
        if x == 1 or x == 2 * i - 1 :
            print('*', end = '')
        else :
            print(' ', end = '')
        x += 1
    print()
```

Output

```
*
* *
*   *
*     *
*   *
* *
*
```

25a. Write programs using nested loops to produce the following patterns:

A
A B
A B C
A B C D
A B C D E
A B C D E F

Solution

```
n = 6
for i in range(n) :
    t = 65
    for j in range(i + 1) :
        print(chr(t), end = ' ')
        t += 1
    print()
```

Output

```
A
A B
A B C
A B C D
A B C D E
A B C D E F
```

25b. Write programs using nested loops to produce the following patterns:

A
B B
C C C
D D D D
E E E E E

Solution

```
n = 5
t = 65

for i in range(n) :
    for j in range(i + 1) :
        print(chr(t), end = ' ')
    t += 1
    print()
```

Output

```
A
B B
C C C
D D D D
E E E E E
```

25c. Write programs using nested loops to produce the following patterns:

0
2 2
4 4 4
6 6 6 6
8 8 8 8 8

Solution

```
for i in range(0, 10, 2):
    for j in range(0, i + 1, 2) :
        print(i, end = ' ')
    print()
```

Output

```
0
2 2
4 4 4
6 6 6 6
8 8 8 8 8
```

25d. Write programs using nested loops to produce the following patterns:

2
4 4
6 6 6
8 8 8 8

Solution

```
for i in range(2, 10, 2) :
    for j in range(2, i + 1, 2) :
        print(i, end = ' ')
    print()
```

Output

```
2
4 4
6 6 6
8 8 8 8
```

26. Write a program using nested loops to produce a rectangle of *'s with 6 rows and 20 *'s per row.

Solution

```
for i in range(6) :
```

```
    for j in range(20) :
        print('*', end = '')
    print()
```

Output

```
********************
********************
********************
********************
********************
********************
```

27. Given three numbers A, B and C, write a program to write their values in an ascending order. For example, if A = 12, B = 10, and C = 15, your program should print out:
Smallest number = 10
Next higher number = 12
Highest number = 15

Solution

```
a = int(input("Enter first number: "))
b = int(input("Enter second number: "))
c = int(input("Enter third number: "))

if a < b and a < c :
    small = a
    if b < c :
        middle = b
        large = c
    else :
        middle = c
        large = b
elif b < a and b < c :
    small = b
    if a < c :
        middle = a
        large = c
    else :
        middle = c
        large = a
else :
    small = c
    if a < b :
        middle = a
        large = b
    else :
        middle = b
        large = a


print("Smallest number =", small)
print("Next higher number =", middle)
print("Highest number =", large)
```

Output

```
Enter first number: 10
Enter second number: 5
```

```
Enter third number: 15
Smallest number = 5
Next higher number = 10
Highest number = 15
```

28. Write a Python script to input temperature. Then ask them what units, Celsius or Fahrenheit, the temperature is in. Your program should convert the temperature to the other unit. The conversions are:
F = 9/5C + 32 and C = 5/9 (F 32).

Solution

```
temp = float(input("Enter Temperature: "))
unit = input("Enter unit('C' for Celsius or 'F' for Fahrenheit): ")

if unit == 'C' or unit == 'c' :
    newTemp = 9 / 5 * temp + 32
    print("Temperature in Fahrenheit =", newTemp)
elif unit == 'F' or unit == 'f' :
    newTemp = 5 / 9 * (temp - 32)
    print("Temperature in Celsius =", newTemp)
else :
    print("Unknown unit", unit)
```

Output

```
Enter Temperature: 38
Enter unit('C' for Celsius or 'F' for Fahrenheit): C
Temperature in Fahrenheit = 100.4
```

29. Ask the user to enter a temperature in Celsius. The program should print a message based on the temperature:

- If the temperature is less than -273.15, print that the temperature is invalid because it is below absolute zero.
- If it is exactly -273.15, print that the temperature is absolute 0.
- If the temperature is between -273.15 and 0, print that the temperature is below freezing.
- If it is 0, print that the temperature is at the freezing point.
- If it is between 0 and 100, print that the temperature is in the normal range.
- If it is 100, print that the temperature is at the boiling point.
- If it is above 100, print that the temperature is above the boiling point.

Solution

```
temp = float(input("Enter Temperature in Celsius: "))

if temp < -273.15 :
    print("Temperature is invalid as it is below absolute zero")
elif temp == -273.15 :
    print("Temperature is absolute zero")
elif -273.15 <= temp < 0:
    print("Temperature is below freezing")
elif temp == 0 :
    print("Temperature is at the freezing point")
elif 0 < temp < 100:
    print("Temperature is in the normal range")
elif temp == 100 :
    print("Temperature is at the boiling point")
else :
    print("Temperature is above the boiling point")
```

Output

```
Enter Temperature in Celsius: -273.15
Temperature is absolute zero
```

30. Write a program to display all of the integers from 1 up to and including some integer entered by the user followed by a list of each number's prime factors. Numbers greater than 1 that only have a single prime factor will be marked as prime.
For example, if the user enters 10 then the output of the program should be:
Enter the maximum value to display: 10
1 = 1
2 = 2 (prime)
3 = 3 (prime)
4 = 2x2
5 = 5 (prime)
6 = 2x3
7 = 7 (prime)
8 = 2x2x2
9 = 3x3
10 = 2x5

Solution

```
import math

n = int(input("Enter an integer: "))

for i in range(1, n + 1) :
    if i == 1:
        print("1 = 1")
    else :
        print(i, "=", end=' ')
        c = 0
        for j in range(1, i + 1) :
            if i % j == 0:
                c += 1
        if c == 2:
            print(i, "(prime)", end = '')
            print()
        else :
            t = i
            while t % 2 == 0 :
                print("2", end='x')
                t = t // 2
            k = 3
            x = math.ceil(math.sqrt(t)) + 1
            while k <= x :
                while (t % k == 0) :
                    print(k, end='x')
                    t = t // k
                k += 2

            if t > 2 :
                print(t, end='x')

            print()
```

Output

```
Enter an integer: 10
1 = 1
2 = 2 (prime)
3 = 3 (prime)
4 = 2x2x
5 = 5 (prime)
6 = 2x3x
7 = 7 (prime)
8 = 2x2x2x
9 = 3x3x
10 = 2x5x
```

1. Write a Python Program to Find the Smallest Divisor of an Integer other than 1.

2. Write a Python Program to Count the Number of Digits in a Number as well as in a String.

3. Write a Python Program to Check whether a Number is a Palindrome or not.

4. Write a Python Program to print all Integers that Aren't Divisible by Either 2 or 3.

5. Write a Python Program to read a Number n and find **1+2+…..+n=?**

6. Write a Python Program to Read a Number n and Print the Natural Numbers Summation Pattern as given below.

1 =1
1+2 =3
1+2+3 =6
1+2+3+4 =10
1+2+3+4+5 =15

7. Write a Python Program to convert the Binary number to its equivalent Decimal Number using function.

8. Write a Python Program to Print all Prime Numbers in a given Range.

9. Write a Python Program to Check if a Date (**inputted in DD-MM-YYYY format**) is Valid or not.

10. Write a Python Program to Compute Simple Interest with all the Required Values.

11. Write a Python Program to Check Whether a Given Year is a Leap Year or not.

12. Write a Python Program to Read Height in Centimetres and then Convert the Height to Feet and Inches.

13. Write a Python Program to Take the Temperature in Celsius and Covert it to the equivalent Fahrenheit.

14. Write a Python Program to print the Prime Factors of an Integer.

15. Write a Python Program to generate all the Divisors of an Integer.

16. Write a Python Program to Print Multiplication Table of a Given Number.

19. Write a Python Program to Print Largest Even and Largest Odd Number in a List.

20. Write a Python Program to Form an Integer that has the Number of Digits at Ten's Place and the Least Significant Digit of the Entered Integer at One's Place.

21. Write a Python Program to Find Those Numbers which are Divisible by 7 and Multiple of 5 in a Given Range of Numbers.

22. Write a Python Program to Check if a Number is an Armstrong Number.

23. Write a Python Program to Print the Pascal's triangle for n number of rows given by the user.

24. Write a Python Program to Check if a Number is a Perfect Number.

25. Write a Python Program to Check if a Number is a Strong Number.

26. Write a Python Program to Find the LCM of Two Numbers.

27. Write a Python Program to Find the GCD of Two Numbers.

28. Write a Python Program to Compute a Polynomial Equation given that the Coefficients of the Polynomial are stored in a List.

29. Write a Python Program to Check If Two Numbers is Amicable Numbers.

30. Write a Python Program to Find the Area of a Triangle Given All Three Sides.

31. Write a Python Program to Print Sum of Negative Numbers, Positive Even Numbers and Positive Odd numbers in a List.

String related problems:

Question 1

Write a program to count the number of times a character occurs in the given string.

Solution

```
str = input("Enter the string: ")
ch = input("Enter the character to count: ");
c = str.count(ch)
print(ch, "occurs", c, "times")
```

Output

```
Enter the string: KnowledgeBoat
Enter the character to count: e
e occurs 2 times
```

Question 2

Write a program which replaces all vowels in the string with '*'.

Solution

```
str = input("Enter the string: ")
newStr = ""
for ch in str :
    lch = ch.lower()
    if lch == 'a' \
```

```
       or lch == 'e' \
       or lch == 'i' \
       or lch == 'o' \
       or lch == 'u' :
        newStr += '*'
    else :
        newStr += ch
print(newStr)
```

Output

```
Enter the string: Computer Studies
C*mp*t*r St*d**s
```

Question 3

Write a program which reverses a string and stores the reversed string in a new string.

Solution

```
str = input("Enter the string: ")
newStr = ""
for ch in str :
    newStr = ch + newStr
print(newStr)
```

Output

```
Enter the string: computer studies
seiduts retupmoc
```

Question 4

Write a program that prompts for a phone number of 10 digits and two dashes, with dashes after the area code and the next three numbers. For example, 017-555-1212 is a legal input. Display if the phone number entered is valid format or not and display if the phone number is valid or not (i.e., contains just the digits and dash at specific places.)

Solution

```
phNo = input("Enter the phone number: ")
length = len(phNo)
if length == 12 \
    and phNo[3] == "-" \
    and phNo[7] == "-" \
    and phNo[:3].isdigit() \
    and phNo[4:7].isdigit() \
    and phNo[8:].isdigit() :
    print("Valid Phone Number")
else :
    print("Invalid Phone Number")
```

Output

```
Enter the phone number: 017-555-1212
Valid Phone Number

====================================

Enter the phone number: 017-5A5-1212
Invalid Phone Number
```

Question 5

Write a program that should do the following :

- prompt the user for a string
- extract all the digits from the string
- If there are digits:
    - sum the collected digits together
    - print out the original string, the digits, the sum of the digits
- If there are no digits:
    - print the original string and a message "has no digits"

Sample

- given the input : abc123
  prints abc123 has the digits 123 which sum to 6
- given the input : abcd
  prints abcd has no digits

Solution

```
str = input("Enter the string: ")
sum = 0
digitStr = ''
for ch in str :
    if ch.isdigit() :
        digitStr += ch
        sum += int(ch)
if not digitStr :
    print(str, "has no digits")
else :
    print(str, "has the digits", digitStr, "which sum to", sum)
```

Output

```
Enter the string: abc123
abc123 has the digits 123 which sum to 6

====================================

Enter the string: KnowledgeBoat
KnowledgeBoat has no digits
```

Question 6

Write a program that should prompt the user to type some sentence(s) followed by "enter". It should then print the original sentence(s) and the following statistics relating to the sentence(s) :

- Number of words
- Number of characters (including white-space and punctuation)
- Percentage of characters that are alphanumeric

Hints

- Assume any consecutive sequence of non-blank characters is a word.

Solution

```
str = input("Enter a few sentences: ")
length = len(str)
spaceCount = 0
alnumCount = 0
```

```
for ch in str :
    if ch.isspace() :
        spaceCount += 1
    elif ch.isalnum() :
        alnumCount += 1

alnumPercent = alnumCount / length * 100

print("Original Sentences:")
print(str)

print("Number of words =", (spaceCount + 1))
print("Number of characters =", (length + 1))
print("Alphanumeric Percentage =", alnumPercent)
```

Output

```
Enter a few sentences: Python was conceived in the late 1980s by
Guido van Rossum at Centrum Wiskunde & Informatica (CWI) in the
Netherlands. Its implementation began in December 1989. Python 3.0
was released on 3 December 2008.
Original Sentences:
Python was conceived in the late 1980s by Guido van Rossum at
Centrum Wiskunde & Informatica (CWI) in the Netherlands. Its
implementation began in December 1989. Python 3.0 was released on 3
December 2008.
Number of words = 34
Number of characters = 206
Alphanumeric Percentage = 80.48780487804879
```

Question 7

Write a Python program as per specifications given below:

- Repeatedly prompt for a sentence (string) or for 'q' to quit.
- Upon input of a sentence s, print the string produced from s by converting each lower case letter to upper case and each upper case letter to lower case.
- All other characters are left unchanged.

For example,

Please enter a sentence, or 'q' to quit : This is the Bomb!

tHIS IS THE bOMB!

Please enter a sentence, or 'q ' to quit : What's up Doc ???

wHAT'S UP dOC ???

Please enter a sentence, or 'q' to quit : q

Solution

```
while True :
    str = input("Please enter a sentence, or 'q' to quit : ")
    newStr = ""
    if str.lower() == "q" :
        break
    for ch in str :
        if ch.islower() :
            newStr += ch.upper()
```

```
        elif ch.isupper() :
            newStr += ch.lower()
        else :
            newStr += ch
    print(newStr)
```

Output

```
Please enter a sentence, or 'q' to quit : This is the Bomb!
tHIS IS THE bOMB!
Please enter a sentence, or 'q' to quit : What's up Doc ???
wHAT'S UP dOC ???
Please enter a sentence, or 'q' to quit : q
```

Question 8

Write a program that does the following :

- takes two inputs : the first, an integer and the second, a string
- from the input string extract all the digits, in the order they occurred, from the string.
    - if no digits occur, set the extracted digits to 0
- add the integer input and the digits extracted from the string together as integers
- print a string of the form :
  "integer_input + string_digits = sum"

For example :

For inputs 12, 'abc123' → '12 + 123 = 135'

For inputs 20, 'a5b6c7' → '20 + 567 =587'

For inputs 100, 'hi mom' → '100 + 0 = 100'

Solution

```
num = int(input("Enter an integer: "))
str = input("Enter the string: ")

digitsStr = ''
digitsNum = 0;

for ch in str :
    if ch.isdigit() :
        digitsStr += ch

if digitsStr :
    digitsNum = int(digitsStr)

print(num, "+", digitsNum, "=", (num + digitsNum))
```

Output

```
Enter an integer: 12
Enter the string: abc123
12 + 123 = 135

====================================

Enter an integer: 20
Enter the string: a5b6c7
20 + 567 = 587
```

```
====================================

Enter an integer: 100
Enter the string: hi mom
100 + 0 = 100
```

Question 9

Write a program that takes two strings from the user and displays the smaller string in single line and the larger string as per this format :

```
1st letter                  last letter
    2nd letter          2nd last letter
        3rd letter  3rd last letter
```

For example,

if the two strings entered are Python and PANDA then the output of the program should be :

```
PANDA
P          n
  y      o
    t  h
```

Solution

```
str1 = input("Enter first string: ")
str2 = input("Enter second string: ")

small = str1
large = str2

if len(str1) > len(str2) :
    large = str1
    small = str2

print(small)

lenLarge = len(large)
for i in range(lenLarge // 2) :
    print(' ' * i, large[i], ' ' * (lenLarge - 2 * i),
large[lenLarge - i - 1], sep='')
```

Output

```
Enter first string: Python
Enter second string: PANDA
PANDA
P      n
 y    o
  t  h
```

Question 10

Write a program to convert a given number into equivalent Roman number (store its value as a string). You can use following guidelines to develop solution for it:

- From the given number, pick successive digits, using %10 and /10 to gather the digits from right to left.

- The rules for Roman Numerals involve using four pairs of symbols for ones and five, tens and fifties, hundreds and five hundreds. An additional symbol for thousands covers all the relevant bases.
- When a number is followed by the same or smaller number, it means addition. "II" is two 1's = 2. "VI" is 5 + 1 = 6.
- When one number is followed by a larger number, it means subtraction. "IX" is 1 before 10 = 9. "IIX isn't allowed, this would be "VIII". For numbers from 1 to 9, the symbols are "I" and "V", and the coding works like this. "I" , "II", "III", "IV", "V", "VI", "VII", "VIII", "IX".
- The same rules work for numbers from 10 to 90, using "X" and "L". For numbers from 100 to 900, using the symbols "C" and "D". For numbers between 1000 and 4000, using "M".

Here are some examples. 1994 = MCMXCIV, 1956 = MCMLVI, 3888= MMMDCCCLXXXVIII

Solution

```
n = int(input("Enter the number: "))
num = (1000, 900,  500, 400, 100,  90, 50,  40, 10,  9,   5,  4,
1)
rom = ('M',  'CM', 'D', 'CD','C',
'XC','L','XL','X','IX','V','IV','I')

result = ''

for i in range(len(num)) :
    count = int(n / num[i])
    result += str(rom[i] * count)
    n -= num[i] * count

print(result)
```

Output

```
Enter the number: 1994
MCMXCIV

====================================

Enter the number: 1956
MCMLVI

====================================

Enter the number: 3888
MMMDCCCLXXXVIII
```

Question 11

Write a program that asks the user for a string (only single space between words) and returns an estimate of how many words are in the string. (Hint. Count number of spaces)

Solution

```
str = input("Enter a string: ")
count = 0
```

```
for ch in str :
    if ch.isspace() :
        count += 1
print("No of words =", (count + 1))
```

Output

```
Enter a string: Python was conceived in the late 1980s by Guido van
Rossum at Centrum Wiskunde & Informatica (CWI) in the Netherlands.
No of words = 20
```

Question 12

Write a program to input a formula with some brackets and checks, and prints out if the formula has the same number of opening and closing parentheses.

Solution

```
str = input("Enter a formula: ")
count = 0

for ch in str :
    if ch == '(' :
        count += 1
    elif ch == ')' :
        count -= 1

if count == 0 :
    print("Formula has same number of opening and closing
parentheses")
else :
    print("Formula has unequal number of opening and closing
parentheses")
```

Output

```
Enter a formula: s(s-a)(s-b)(s-c)
Formula has same number of opening and closing parentheses

====================================

Enter a formula: s((s-a)(s-b)(s-c)
Formula has unequal number of opening and closing parentheses
```

Question 13

Write a program that inputs a line of text and prints out the count of vowels in it.

Solution

```
str = input("Enter a string: ")
count = 0

for ch in str :
    lch = ch.lower()
    if lch == 'a' \
       or lch == 'e' \
       or lch == 'i' \
       or lch == 'o' \
       or lch == 'u' :
       count += 1
```

```
print("Vowel Count =", count)
```
Output
```
Enter a string: Internet of Things
Vowel Count = 5
```

Question 14

Write a program to input a line of text and print the biggest word (length wise) from it.

Solution
```
str = input("Enter a string: ")
words = str.split()
longWord = ''

for w in words :
    if len(w) > len(longWord) :
        longWord = w

print("Longest Word =", longWord)
```
Output
```
Enter a string: TATA FOOTBALL ACADEMY WILL PLAY AGAINST MOHAN BAGAN
Longest Word = FOOTBALL
```

Question 15

Write a program to input a line of text and create a new line of text where each word of input line is reversed.

Solution
```
str = input("Enter a string: ")
words = str.split()
newStr = ""

for w in words :
    rw = ""
    for ch in w :
        rw = ch + rw
    newStr += rw + " "

print(newStr)
```
Output
```
Enter a string: Python is Fun
nohtyP si nuF
```

List related problems:

1. Write a program to increment the elements of a list with a number.

Solution
```
lst = eval(input("Enter a list: "))
print("Existing list is:", lst)

n = int(input("Enter a number: "))

for i in range(len(lst)):
```

```
    lst[i] += n

print("List after increment:", lst)
```

Output

```
Enter a list: [1, 2, 3, 4, 5]
Existing list is: [1, 2, 3, 4, 5]
Enter a number: 10
List after increment: [11, 12, 13, 14, 15]
```

2. Write a program that reverses a list of integers (in place).

Solution

```
l = eval(input("Enter a list: "))
print("Original list:", l)
l.reverse()
print("Reversed list:", l)
```

Output

```
Enter a list: [1, 2, 3, 4, 5]
Original list: [1, 2, 3, 4, 5]
Reversed list: [5, 4, 3, 2, 1]
```

Question 3

Write a program that inputs two lists and creates a third, that contains all elements of the first followed by all elements of the second.

Solution

```
l1 = eval(input("Enter first list: "))
l2 = eval(input("Enter second list: "))
l3 = l1 + l2
print("Joined List:", l3)
```

Output

```
Enter first list: [1, 2, 3, 4, 5]
Enter second list: [11, 12, 13, 14, 15]
Joined List: [1, 2, 3, 4, 5, 11, 12, 13, 14, 15]
```

Question 4

Ask the user to enter a list containing numbers between 1 and 12. Then replace all of the entries in the list that are greater than 10 with 10.

Solution

```
l = eval(input("Enter list having numbers between 1 & 12: "))

for i in range(len(l)):
    if l[i] > 10:
        l[i] = 10

print("List after removing numbers greater than 10:")
print(l)
```

Output

```
Enter list having numbers between 1 & 12: [1, 3, 15, 8, 20]
List after removing numbers greater than 10:
[1, 3, 10, 8, 10]
```

Question 5

Ask the user to enter a list of strings. Create a new list that consists of those strings with their first characters removed.

Solution

```
l1 = eval(input("Enter a list of strings: "))
l2 = []
```

```
for i in range(len(l1)):
    l2.append(l1[i][1:])

print("List after removing first characters:")
print(l2)
```

Output

```
Enter a list of strings: ["red", "green", "blue", "pink", "cyan"]
List after removing first characters:
['ed', 'reen', 'lue', 'ink', 'yan']
```

Question 6

Write a program to check if a number is present in the list or not. If the number is present, print the position of the number. Print an appropriate message if the number is not present in the list.

Solution

```
l = eval(input("Enter list: "))
n = int(input("Enter number to search: "))

if n in l:
    print(n, "found at index", l.index(n))
else :
    print(n, "not found in list")
```

Output

```
Enter list: [1, 3, 15, 8, 20]
Enter number to search: 15
15 found at index 2

====================================

Enter list: [1, 3, 15, 8, 20]
Enter number to search: 25
25 not found in list
```

Question 7a

Create the following lists using a for loop:
A list consisting of the integers 0 through 49.

Solution

```
l = []

for i in range(50):
    l.append(i)

print("List with integers from 0 to 49:")
print(l)
```

Output

```
List with integers from 0 to 49:
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28,
29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49]
```

Question 7b

Create the following lists using a for loop:
A list containing the squares of the integers 1 through 50.

Solution

```
l = []
```

```
for i in range(1, 51):
    l.append(i * i)

print("List with square of integers from 1 to 50:")
print(l)
```

Output

List with square of integers from 1 to 50:
[1, 4, 9, 16, 25, 36, 49, 64, 81, 100, 121, 144, 169, 196, 225, 256, 289, 324, 361, 400, 441, 484, 529, 576, 625, 676, 729, 784, 841, 900, 961, 1024, 1089, 1156, 1225, 1296, 1369, 1444, 1521, 1600, 1681, 1764, 1849, 1936, 2025, 2116, 2209, 2304, 2401, 2500]

Question 7c

Create the following lists using a for loop:
The list ['a','bb','ccc','dddd', . . .] that ends with 26 copies of the letter z.

Solution

```
l = []

for i in range(1, 27):
    l.append(chr(i + 96) * i)

print("Created List:")
print(l)
```

Output

Created List:
['a', 'bb', 'ccc', 'dddd', 'eeeee', 'ffffff', 'ggggggg', 'hhhhhhhh', 'iiiiiiiii', 'jjjjjjjjj', 'kkkkkkkkkk', 'lllllllllll', 'mmmmmmmmmmmmm', 'nnnnnnnnnnnnnn', 'ooooooooooooooo', 'pppppppppppppppp', 'qqqqqqqqqqqqqqqqq', 'rrrrrrrrrrrrrrrrr', 'ssssssssssssssssss', 'ttttttttttttttttttt', 'uuuuuuuuuuuuuuuuuuuu', 'vvvvvvvvvvvvvvvvvvvvv', 'wwwwwwwwwwwwwwwwwwwww', 'xxxxxxxxxxxxxxxxxxxxxxx', 'yyyyyyyyyyyyyyyyyyyyyyyy', 'zzzzzzzzzzzzzzzzzzzzzzzzzz']

Question 8

Write a program that takes any two lists L and M of the same size and adds their elements together to form a new list N whose elements are sums of the corresponding elements in L and M. For instance, if L = [3, 1, 4] and M = [1, 5, 9], then N should equal [4,6,13].

Solution

```
print("Enter two lists of same size")
L = eval(input("Enter first list(L): "))
M = eval(input("Enter second list(M): "))
N = []

for i in range(len(L)):
    N.append(L[i] + M[i])

print("List N:")
print(N)
```

Output

Enter two lists of same size
Enter first list(L): [3, 1, 4]
Enter second list(M): [1, 5, 9]
List N:
[4, 6, 13]

Question 9

Write a program rotates the elements of a list so that the element at the first index moves to the second index, the element in the second index moves to the third index, etc., and the element in the last index moves to the first index.

Solution

```
l = eval(input("Enter the list: "))
print("Original List")
print(l)

l = l[-1:] + l[:-1]

print("Rotated List")
print(l)
```

Output

```
Enter the list: [8, 10, 13, 25, 7, 11]
Original List
[8, 10, 13, 25, 7, 11]
Rotated List
[11, 8, 10, 13, 25, 7]
```

Question 10

Write a program that reads the n to display nth term of Fibonacci series.

The Fibonacci sequence works as follows:

- element 0 has the value 0
- element 1 has the value 1
- every element after that has the value of the sum of the two preceding elements

The beginning of the sequence looks like:

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, ...

The program prompts for element and prints out the value of that element of the Fibonacci sequence.

Thus:

- input 7, produces 13
- input 9, produces 34

Hints:

A Don't try to just type out the entire list. It gets big very fast. Element 25 is 75205. Element 100 is 354224848179261915075. So keep upper limit of n to 20.

Solution

```
n = int(input("Enter n: "))

if (n > 20):
    print("n should be less than or equal to 20")
else :
    a = 0
    b = 1
    c = a + b
    for i in range(3, n + 1):
        a = b
        b = c
        c = a + b

    print(n, "term of Fibonacci series =", c)
```

Output

```
Enter n: 7
7 term of Fibonacci series = 13
```

```
====================================

Enter n: 9
9 term of Fibonacci series = 34
====================================
Enter n: 25
n should be less than or equal to 20
```

Question 11a

Write programs as per following specifications:
'''Print the length of the longest
string in the list of strings str_list.
Precondition : the list will contain
at least one element.'''

Solution

```
l = eval(input("Enter list of strings: "))
largeIdx = 0
largeLen = 0

for i in range(len(l)):
    length = len(l[i])
    if length > largeLen:
        largeLen = length
        largeIdx = i

print("Longest String:", l[largeIdx])
```

Output

```
Enter list of strings: ["apple", "orange", "pear", "strawberry", "kiwi"]
Longest String: strawberry
```

Question 11b

Write programs as per following specifications:
'''L is a list of numbers. Print a new list where each element is the corresponding element of list L summed with number num.'''

```
Solution
l1 = eval(input("Enter list of numbers: "))
num = int(input("Enter the number to sum with (num): "))

l2 = []

for i in l1:
    l2.append(i + num)

print("New list:")
print(l2)
```

Output

```
Enter list of numbers: [10, 20, 30, 40, 50]
Enter the number to sum with (num): 15
New list:
[25, 35, 45, 55, 65]
```

Question 12

Write a program to read two lists num and denum which contain the numerators and denominators of same fractions at the respective indexes. Then display the smallest fraction along with its index.

Solution

```
num = eval(input("Enter numerators list: "))
denum = eval(input("Enter denominators list: "))

small = 0.0
smallIdx = 0

for i in range(len(num)):
    t = num[i] / denum[i]
    if t < small:
        small = t
        smallIdx = i

print("Smallest Fraction =", num[smallIdx], "/", denum[smallIdx])
print("Index of Smallest Fraction =", smallIdx)
```

Output

```
Enter numerators list: [1, 3, 1, 7, 3]
Enter denominators list: [2, 4, 4, 13, 8]
Smallest Fraction = 1 / 2
Index of Smallest Fraction = 0
```

Question 13

Write a program to display the maximum and minimum values from the specified range of indexes of list.

Solution

```
l = eval(input("Enter the list: "))
start = int(input("Enter start index: "))
stop = int(input("Enter stop index: "))

slice = l[start : stop + 1]
mx = max(slice)
mi = min(slice)

print("Maximum =", mx)
print("Minimum =", mi)
```

Output

```
Enter the list: [89, 42, 12, 56, 35, 2, 8, 7, 13, 69]
Enter start index: 3
Enter stop index: 8
Maximum = 56
Minimum = 2
```

Question 14

Write a program to move all duplicate values in a list to the end of the list.

Solution

```
l = eval(input("Enter the list: "))
dedup = []
dup = []
for i in l:
    if i in dedup:
        dup.append(i)
    else:
```

```
        dedup.append(i)

l = dedup + dup

print("Modified List:")
print(l)
```

Output

```
Enter the list: [20, 15, 18, 15, 7, 18, 12, 13, 7]
Modified List:
[20, 15, 18, 7, 12, 13, 15, 18, 7]
```

Question 15

Write a program to compare two equal sized lists and print the first index where they differ.

Solution

```
print("Enter two equal sized lists")
l1 = eval(input("Enter first list: "))
l2 = eval(input("Enter second list: "))

for i in range(len(l1)):
    if l1[i] != l2[i]:
        print("Lists differ at index", i)
        break;
else:
    print("Lists are equal")
```

Output

```
Enter two equal sized lists
Enter first list: [80, 60, 50, 40, 30]
Enter second list: [80, 60, 55, 42, 30]
Lists differ at index 2

====================================

Enter two equal sized lists
Enter first list: [80, 60, 50, 40, 30]
Enter second list: [80, 60, 50, 40, 30]
Lists are equal
```