**Data Analysis and Machine-Learning**

Chapter 1-5:

Introductory Algorithm-Making Problems



Problem (4)

**1. Task Definition**

Create a python algorithm that displays the output of the sum of given set of integer inputs, X and Y. The set of input values, X and Y, are provided line by line, separated by a single space. Two 0 values are given on the last line (0 0).

**2. Sample Input:**

14 12

4 7

5 8

8 7

3 2

8 8

0 0

**3. Expected Output:**

26

11

13

15

5

16

**Hints:**

1. ***While*** loop expression iterates given command, of which the loop process can be stopped manually by utilizing ***Break*** expression.

**Sample Answer:**

1.

while True:

    X, Y = input().split()

    X, Y = int(X), int(Y)

    if X == 0 and Y == 0:

        break

    print(X+Y)

2.

while True:

      a,b = map(int, input().split())

      if a==0 and b==0:

            break

      else:

            print(a+b)

Problem (5)

**1. Task Definition**

Create a python algorithm that displays the output of the sum of given set of integer inputs, X and Y. the total number of sets of X and Y is provided on the first line. The input values of X and Y are provided from the second line, separated by “,”; and each set of X and Y is separated by lines.

**2. Sample Input:**

14,12

4,7

5,8

8,7

3,2

8,8

**3. Expected Output:**

26

11

13

15

5

16

**Hints:**

1. ***Split()*** expression can come with specific conditions, e.g., split by “,” or space(“ “), etc.

**Sample Answer:**

1.

a = int(input())

for x in range(a):

    b = input().split(',')

    X, Y = int(b[0]), int(b[1])

    print(X+Y)

2.

a = int(input())

for x in range(a):

      b,c = map(int, input().split(','))

      print(b+c)

Problem (5)

**1. Task Definition**

Create a python algorithm that displays the output of the sum of given set of integer inputs, X and Y. the total number of sets of X and Y is provided on the first line. The input values of X and Y are provided from the second line, separated by a single space; and each set of X and Y is separated by lines.

**2. Sample Input:**

6

14 12

4 7

5 8

8 7

3 2

8 8

**3. Expected Output:**

Expected Output for line 1: 14 + 12 = 26

Expected Output for line 2: 4 + 7 = 11

Expected Output for line 3: 5 + 8 = 13

Expected Output for line 4: 8 + 7 = 15

Expected Output for line 5: 3 + 2 = 5

Expected Output for line 6: 8 + 8 = 16

**Hints:**

1. You can utilize either a ***.format*** expression or a ***f”{}”*** expression.

**Sample Answer:**

1.

a = int(input())

for i in range(a):

    X, Y = input().split()

    X, Y = int(X), int(Y)

    print('Expected Output for line {}: {} + {} = {}'.format(i+1, X, Y, X+Y))

2.

a = int(input())

for i in range(a):

      b,c = map(int, input().split())

      print(f'Expected Output for line {i+1}: {b} + {c} = {b+c}')

Problem (6)

**1. Task Definition**

Create a python algorithm that displays the output of the sum of given set of integer inputs, X and Y. the total number of sets of X and Y is provided on the first line. The input values of X and Y are provided from the second line, separated by a single space; and each set of X and Y is separated by lines.

**2. Sample Input:**

6

14 12

4 7

5 8

8 7

3 2

8 8

**3. Expected Output:**

Expected Output for line 1: 14 + 12 = 26

Expected Output for line 2: 4 + 7 = 11

Expected Output for line 3: 5 + 8 = 13

Expected Output for line 4: 8 + 7 = 15

Expected Output for line 5: 3 + 2 = 5

Expected Output for line 6: 8 + 8 = 16

**Hints:**

1. You can utilize either a ***.format*** expression or a ***f”{}”*** expression.

**Sample Answer:**

1.

a = int(input())

for i in range(a):

    X, Y = input().split()

    X, Y = int(X), int(Y)

    print('Expected Output for line {}: {} + {} = {}'.format(i+1, X, Y, X+Y))

2.

a = int(input())

for i in range(a):

      b,c = map(int, input().split())

      print(f'Expected Output for line {i+1}: {b} + {c} = {b+c}')

Problem (8)

**1. Task Definition**

Create a python algorithm that displays the total sum of integers from 1 to X on the output, when X is given as the input.

**2. Sample Input:**

10

**3. Expected Output:**

55

**Hints:**

1. Use ***For*** loop and ***range()***.

**Sample Answer:**

1.

a = int(input())

for x in range(a):

    a = a + x

print(a)

2.

a = int(input())

b = 0

for x in range(1, a+1):

    b = b+x

print(b)

3.

a=int(input())

b=0

for i in range(1, a+1):

      b=b+i

print(b)

Problem (9)

**1. Task Definition**

Create a python algorithm that displays the total sum of integers from 1 to X on the output, when X is given as the input value.

**2. Sample Input:**

10

**3. Expected Output:**

55

**Hints:**

1. Use ***For*** loop and ***range()***.

**Sample Answer:**

1.

a = int(input())

for x in range(a):

    a = a + x

print(a)

2.

a = int(input())

b = 0

for x in range(1, a+1):

    b = b+x

print(b)

3.

a=int(input())

b=0

for i in range(1, a+1):

      b=b+i

print(b)

Problem (10)

**1. Task Definition**

Create a python algorithm that displays the multiplied value of the given inputs only using summation (+) commands, and without using multiplication (\*). Input values are separated by a single space.

**2. Sample Input:**

4 8

**3. Expected Output:**

32

**Hints:**

1. Identity element of a summation (addition) is 0

**Sample Answer:**

1.

a = input().split()

X, Y = int(a[0]), int(a[1])

multiply = 0

for \_ in range(Y):

    multiply = multiply + X

print(multiply)

Problem (11)

**1. Task Definition**

Create a python algorithm that displays the multiplied value of the given inputs only using multiplication (\*) commands, and without using exponentiations (\*\*). Input values are separated by a single space.

**2. Sample Input:**

2 4

**3. Expected Output:**

16

**Hints:**

1. Identity element of multiplication is 1

**Sample Answer:**

1.

a = int(input())

b = int(input())

index = 1

for x in range(b):

    index = index \* a

print(index)

Problem (9-2)

**1. Task Definition**

Create a python algorithm that displays the multiplied value of the given inputs only using multiplication (\*) commands, and without using exponentiations (\*\*). Input values are separated by a single space.

**2. Sample Input:**

2 4

**3. Expected Output:**

16

**Hints:**

1. Identity element of multiplication is 1

**Sample Answer:**

1.

a = int(input())

b = int(input())

index = 1

for x in range(b):

    index = index \* a

print(index)

Problem (12)

**1. Task Definition**

In a given sequence (X) consisting of integer values, create a python algorithm that displays only the integer values that are smaller than the given integer value Y. The first line of inputs respectively displays the number of integer values in a sequence X and Y, separated by a single space. The second line of inputs displays the set of integer values in a sequence X.

**2. Sample Input:**

15 11

8 17 15 7 6 3 22 131 75 5 19 1 10 12 11

**3. Expected Output:**

8 7 6 3 5 1 10

**Hints:**

1. ***for*** loop with ***range()*** and ***len()*** functions

**Sample Answers:**

1.

a = input().split()

x, y = int(a[0]), int(a[1])

b = input().split()

for i in range(len(b)):

    b[i] = int(b[i])

for j in range(len(b)):

    if b[j] < y:

        print(b[j], end = ' ')

2.

a,b = map(int, input().split())

c = input().split()

for i in range(len(c)):

      c[i] = int(c[i])

for i in range(len(c)):

      if b > c[i]:

            print(c[i], end = ' ')

Problem (13)

**1. Task Definition**

X number of integers are given without spaces. Create a python algorithm that displays the sum of all integers.

**2. Sample Input:**

19

1758000000000000281

**3. Expected Output:**

32

**Hints:**

1. Set initial base value as 0 for summation via ***for*** loop.

**Sample Answers:**

1.

a = int(input())

b = input()

sum = 0

for x in b:

    sum = sum + int(x)

print(sum)

2.

a = int(input())

b = input()

list\_ = list(b)

init\_ = 0

for i in range(len(list\_)):

      list\_[i] = int(list\_[i])

for j in range(a):

      init\_ = init\_ + list\_[j]

print(init\_)

Problem (14)

**1. Task Definition**

Create a python algorithm that counts the total number of words in a given sentence.

**2. Sample Input:**

Value neutral guidance for tomorrow’s society

**3. Expected Output:**

32

**Hints:**

1. ***strip()***, ***split()***, and ***len()***.

**Sample Answers:**

1.

a = input().split()

print(len(a))

2.

b = input().strip().split(' ')

print(len(b))

Problem (15)

**1. Task Definition**

Integer value X and strings Y are given as inputs, each separated by a single space. Create a python algorithm that iterates each string in Y by X times. For example, the output for “3 xyz” should be xxxyyyzzz.

**2. Sample Input:**

7

5 abcd

8 \*t-b

**3. Expected Output:**

aaaaabbbbbcccccddddd

\*\*\*\*\*\*\*\*tttttttt--------bbbbbbbb

**Hints:**

1. ***for*** loop with split() and range(len()) expressions

**Sample Answers:**

1.

a = int(input())

for \_ in range(a):

    b, c = input().split()

    text = ''

    for x in c:

        text = text + x\*int(b)

    print(text)

2.

a = int(input())

for i in range(a):

      b, c = input().split()

      b = int(b)

      text = ''

      for j in range(len(c)):

        text = text + c[j]\*b

      print(text)

Problem (16)

**1. Task Definition**

Create a python algorithm that generates the minimum and maximum values in a given set of integers. The first line of input displays the number of input integer values. The second line of input displays the input integer values, separated by a single space.

**2. Sample Input:**

11

17 81 55 37 33 11 7 83 54 19 22

**3. Expected Output:**

7 83

**Hints:**

1. For starters, designate initial base value for maximum and minimum, respectively, in a given list of integer values.

**Sample Answers:**

1.

a = int(input())

b = input().split()

for x in range(len(b)):

    b[x] = int(b[x])

min = b[0]

max = b[1]

for x in b:

    if min > x:

        min = x

    if max < x:

        max = x

print(min, max)

2.

a = int(input())

b = input().split()

for i in range(len(b)):

      b[i] = int(b[i])

initial\_max = b[0]

initial\_min = b[0]

for j in range(len(b)):

      if initial\_max < b[j]:

            initial\_max = b[j]

for k in range(len(b)):

      if initial\_min > b[k]:

            initial\_min = b[k]

print(initial\_min, initial\_max)

Problem (17)

**1. Task Definition**

Integers A, B, C, and D are given as inputs, separated by lines. Create a python algorithm that calculates and displays the counts of integers from 0 to 9 in a multiplication value of all A, B, C, and D. For example, if A is 1, B is 2, C is 3, and D is 4, the multiplication value of A,B,C,D is 24. In this case, the counts of integers from 0 to 9 would be: 2 counted once and 4 counted once.

**2. Sample Input:**

12

175

82

33

**3. Expected Output:**

2

0

1

0

0

1

2

0

1

0

**Hints:**

1. Transpose int to str type and create a list from 0 to 9 to count for each index in a list.

**Sample Answers:**

1.

a = input()

b = input()

c = input()

d = input()

i = str(int(a) \* int(b) \* int(c) \* int(d))

j = [0 for \_ in range(0, 10)]

for x in i:

    j[int(x)] = j[int(x)] + 1

for x in j:

    print(x)

2.

a = int(input())

b = int(input())

c = int(input())

d = int(input())

e = str(a\*b\*c\*d)

list\_ = list(e)

for i in range(len(list\_)):

      list\_[i] = int(list\_[i])

f = [0 for \_ in range(10)]

for j in list\_:

      f[j] = f[j] + 1

for k in f:

      print(k)

Problem (18)

**1. Task Definition**

Create a Euclidean algorithm to calculate the GCD (greatest common factor) of the given input integer values.

\* Finding a GCD using Euclidean algorithm goes through the following steps.

**Step 1.** For the given integers a and b; if b is larger than a, set b value as a, and a value as b.

**Step 2.** Set the remainder of a / b as N.

**Step 3.** If N == 0, b is GCD.

**Step 4.** If N != 0, set a value as b, b value as N, and go back to step 2.

**2. Sample Input:**

200

250

**3. Expected Output:**

50

**Hints:**

1. ***While*** loop with ***break*** and ***else*** expressions.

**Sample Answers:**

1.

a = int(input())

b = int(input())

if b>a:

    b, a = a, b

while True:

    n = a%b

    if n == 0:

        break

    else:

        a, b = b, n

print(b)

Problem (19)

**1. Task Definition**

Create times tables for 2 and 3, and display them in a text format of “A x B = C”.

**2. Expected Output:**

2x1=2

2x2=4

2x3=6

2x4=8

2x5=10

2x6=12

2x7=14

2x8=16

2x9=18

3x1=3

3x2=6

3x3=9

3x4=12

3x5=15

3x6=18

3x7=21

3x8=24

3x9=27

**Hints:**

1. ***Nested for loop*** (for loop within a for loop)

**Sample Answers:**

1.

for x in range(2, 4):

    for y in range(1, 10):

        print('{}x{}={}'.format(x, y, x\*y))

Problem (20)

**1. Task Definition**

Create a Python algorithm that calculates and displays the input integer values, X and Y. X and Y are separated by a single space; the sets of X and Y are separated by lines.

**2. Sample Input:**

8 7

11 53

5 1

7 4

81 42

9 9

**3. Expected Output:**

15

64

6

11

123

18

**Hints:**

1. ***While*** loop with ***try*** and ***except*** expressions.

2. EOFError refers to end of file errors.

**Sample Answers:**

1.

while True:

      try:

        a,b = map(int, input().split())

        print(a+b)

      except EOFError:

        break

Problem (21)

**1. Task Definition**

Create a Python algorithm that generates the following shape of stars (\*). Stars should be located within **lists**.

**3. Expected Output:**

['\*', ' ', ' ', ' ', ' ']

['\*', '\*', ' ', ' ', ' ']

['\*', '\*', '\*', ' ', ' ']

['\*', '\*', '\*', '\*', ' ']

['\*', '\*', '\*', '\*', '\*']

**Hints:**

1. ***Nested For loops*** (for loop within a for loop) and list indexing

**Sample Answers:**

1.

star = [[' ' for \_ in range(5)] for \_ in range(5)]

for i in range(5):

      for j in range(i+1):

            star[i][j] = '\*'

print(\*star, sep='\n')

Problem (12)

**1. Task Definition**

Create a Python function that calculates the N number of random integers in a list. Print the randomly generated list and solve by using the defined function.

**2. Sample Output:**

>>> print(list\_)

[6, 9, 9, 5, 10, 2, 4, 7, 1, 5]

>>> print(solve(list\_))

58

**3. Sample Answer:**

1.

def solve(x):

      total = 0

      for i in x:

            total = total + i

      return total

import random as rd

list\_ = [rd.randint(1, 10) for \_ in range(10)]

print(list\_)

print(solve(list\_))

Problem (23)

**1. Task Definition**

Create a Python function that returns (1) True if the input strings are convertible to integers, and (2) False if not convertible to integers. Define the name of the created function as convert().

Use sample inputs “2824” and “2824a” for test.

**2. Sample Output:**

>>> convert('2824')

'true'

>>> convert('2824a')

'false'

**3. Sample Answer:**

1.

def convert(a):

      if (a[0] == '-' or '+') and (a[1:].isdecimal()):

            return 'true'

      elif a[:].isdecimal():

            return 'true'

      else:

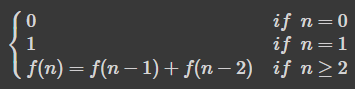
            return 'false'

Problem (24)

**1. Task Definition**

Create a Fibonacci algorithm as a function, and find out the 100th Fibonacci number using the defined function, fibonacci(n).

The recurrence formula for Fibonacci sequence is as follows:



**2. Sample Output:**

>>> fibonacci(100)

354224848179261915075

**2. Hint:**

1. Set the initial base values for Fibonacci sequence as a list, [1, 1].

2. list.append()

**3. Sample Answer:**

1.

def fibonacci(a):

      fibo = [1,1]

      for i in range(0, a):

            fibo.append(fibo[i+1]+fibo[i])

      return fibo[i]

print(fibonacci(100))