1.B) struct

2. C) 1\_no

3. A) in

4.A) Left to Right

5. C) iv – iii – ii – i

6. C) 0.3333…

7. A) x\*\*4\*\*4

8. A) int

9. A) Division and multiplication have same precedence in python & B) Python’s operators’ precedence is based on PEDMAS & D) In case of operators’ having same precedence, the one on the left side is executed first

10. A) abc = 1,000,000 & C) a,b,c = 1000, 2000, 3000 & D) a\_b\_c = 1,000,000

**ANS.11**

* **Lists**

List is a collection which is ordered.

**Lists are mutable (changeable) .**

Allows duplicate members

Brackets used to represent: []

Lists are like arrays declared in other languages.

* **Tuples**

Collection of items which is ordered.

**Tuples are immutable (unchangeable)** .

Brackets used to represent: ()

**Only difference between tuples and lists are that lists can be changed**.

**Tuples are faster than lists** as they are immutable.

* **Sets**

Collection of Unordered and Unindexed items.

**Sets are mutable (changeable)**.

**Does not take duplicate Values**.

Sets are unordered, so you cannot be sure in which order the items will appear.

Brackets used to represent: { }.

**Sets are not faster than lists** however they have a upper hand when it comes to membership testing.

* **Dictionaries**

Key:Value Pair in Python

**A dictionary is a collection which is unordered, changeable and indexed**.

In Python dictionaries are written with curly brackets, and they have keys and values.

Brackets used to represent: {}.

**ANS.12**. **Strings are immutable**

Numbers, strings, and tuples are immutable. Lists, dictionaries, and sets are mutable. Immutability may be used to ensure that an object remains constant throughout your program. The values of mutable objects can be changed at any time and place, whether you expect it or not.

To replace ‘+’ with space in python,

INPUT

a="i+Love+python"

a.replace("+"," ").

OUTPUT

'i Love python'

**ANS.13**. **The ord() function in Python accepts a string of length 1 as an argument and returns the unicode code point representation of the passed argument**. **For example ord('B') returns 66** which is a unicode code point value of character ‘B’.

**To check the data type of a variable** we can write this code **type(variable\_name).**This will give the data type of that variable,**for example** we have to check data type of a, so we will write **type(a)**.

**ANS.14**.INPUT

from math import sqrt

print("Quadratic function : (a \* x^2) + b\*x + c")

a = float(input("a: "))

b = float(input("b: "))

c = float(input("c: "))

r = b\*\*2 - 4\*a\*c

if r > 0:

num\_roots = 2

x1 = (((-b) + sqrt(r))/(2\*a))

x2 = (((-b) - sqrt(r))/(2\*a))

print("There are 2 roots: %f and %f" % (x1, x2))

elif r == 0:

num\_roots = 1

x = (-b) / 2\*a

print("There is one root: ", x)

else:

num\_roots = 0

print("No roots, discriminant < 0.")

NOTE:

{this (input("a: ")and similarly for b,c will let us put values of a,b and c.

OUTPUT

I entered the value of a as 1 b as 5 c as 6 and then I pressed enter.

Quadratic function : (a \* x^2) + b\*x + c

a: 1

b: 5

c: 6

There are 2 roots: -2.000000 and -3.000000

**ANS.15**. INPUT

print("sum of first n natural numbers :n\*(n+1)/2 ")

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

print("sum of first n natural number is: ")

n\*(n+1)/2

OUTPUT

sum of first n natural numbers :n\*(n+1)/2

n: 20 (NOTE:I entered the value of n as 20 and pressed enter)

sum of first n natural number is:

210.0