Answers of Question Bank
Class XI
Computer Science
(Book Reference : Sumita Arora)

Chapter – 1: GETTING STARTED WITH PYTHON
Very Short answer Type Questions

Q.1   When was Python released?

    Ans- Python was released in February 1991.

Q.2   Who developed Python?

    Ans- Python was developed by Guido Van Rossum.

Q.3   Which two languages contributed to Python as a Programming Language?

    Ans- ABC language and Modula 3.

Q.4   Is Python an Object Oriented Language?

    Ans- Yes, Python is an object oriented language.

Q.5   Python got its name from which show?

    Ans- Python got its name from famous BBC comedy show "Monty Python's Flying Circus".

Short Answer Type Questions

Q.1   „Python is an interpreted language". What does it mean to you?
Ans- It means that the Python installation interprets and executes the code line by line at a time.

Q.2   What does a cross platform language mean?
Ans- it means a language can run equally on variety of platforms-Windows, Linux/UNIX, Macintosh,

Supercomputers, Smart phones etc.

Q.3   Python is a Free and Open Source language. What do you understand by this feature?
Ans- It means Python is freely available without any cost. Its source code is also available. One can modify, improve/extend an open source software.

Q.4   In how many ways, can you work in Python?
Ans-In two ways we can work in Python-
        (i) Interactive mode
        (ii) Script Mode

Q.5   What is the difference between Interactive mode and Script Mode in Python?

    Ans- In interactive mode, one command can run at a time and commands are not saved. Whereas in Script mode, we can save all the commands in the form of a program file and can see output of all lines together.

Q.6   What are the advantages of Python?

Ans- Advantages-
i. Easy to use OO Language.
ii. Expressive Language.
iii. Interpreted Language.
iv. It is complete.
v. Cross-Platform Language.
vi. Free and Open Source.
vii. Variety of Usage/ Applications

Q.7   What are the limitations of Python?

Ans- limitations-
i. Not the fastest language.

ii. Lesser Libraries than C, Java, Perl.

iii. Not strong on Type Binding.

iv. Not easily convertible.

Skill Based Questions

Q.1 **Which of the following are not valid strings in Python?**

(a)‖ Hello‖   (b) ‗Hello'   (c)‖ Hello'   (d) ‗Hello‖   (e) {Hello}

Ans- String (c) , (d) and (e) are not valid strings.

Q.2 **Write Instructions in python to get the following result:**

Q.3 **(Do it in both interactive mode and script mode)**

I am a student of KV Barabanki

I live in Barabanki

And I love Barabanki.

Barabanki is 20 KM away from Lucknow

This Place is famous for Dewa Sharif

Ans- **In Interactive Mode-**

>>> print("I am a student of KV Barabanki \nI live in Barabanki \nAnd I love Barabanki\n\n Barabanki is 20 KM away from Lucknow\n This Place is famous for Dewa Sharif")

In Script Mode-

print("I am a student of KV Barabanki \n I live in Barabanki \nAnd I love Barabanki\n\n Barabanki is 20 KM away from Lucknow\n This Place is famous for Dewa Sharif")

(save this code to a file and then run the file)

Chapter – 2: PYTHON FUNDAMENTALS

Very Short answer Type Questions

Q.1 **What is None literal in Python?**

Ans: Python has one special literal, which is **None**. The None literal is used to indicate absence of value. It is also used to indicate the end of lists in Python. It means ―There is nothing here‖ .

Q.2 **What is the error in following code: x, y =7 ?**

Ans: The following error comes - 'int' object is not iterable. Which means an integer object i.e. cannot be repeated for x and y. one more integer object is required after 7.

Q.3 **what will the following code do: a=b=18 ?**

Ans: This code will assign 18 to a and b both.

Q.4 **Following code is creating problem X = 0281, find reason.**

Ans: 0281 is an invalid token.

Q.5 **Find the error in the following code:**

(a) *temp=90*
      Print temp

(b) *a=12*
      b = a + b
      print(a And b)

(c) *print("x="x)*

(d) *a, b, c=2, 8, 4*
      print(a, b, c)
      c, b, a = a, b, c
      print(a; b; c)

(e) *x = 23*
      4 = x

(f) *else = 21-4*

Ans: (a)   Missing parentheses in call to 'print'.
(b)   Name ‗b' is not defined.
(c)   Invalid Syntax.
(d)   Invalid Syntax in second print statement.
(e)   can't assign to literal in second line.
(f)   Invalid Syntax.

Q.6 **Find the error in the following code:**

(a) y = x +5
    print(x,y)

(b) a=input("Value: ")
    b = a/2
    print(a, b)

(c) print(x = y = 5)

Ans: (a)   Name 'x' is not defined.

    (b)   Unsupported operand type(s) for /: 'str' and 'int'.

    (c)   Invalid Syntax.

Short Answer Type Questions

Q.1 **What is the difference between a keyword and an identifier?**

Ans: Difference between Keyword and Identifier: Every language has keywords and identifiers, which are only understood by its compiler. Keywords are predefined reserved words, which possess special meaning. An identifier is a unique name given to a particular variable, function or label of class in the program.

Q.2 **What are literals in Python? How many types of Literals allowed in Python?**

Ans: **Literals:** Python comes with some built-in objects. Some are used so often that Python has a quick way to make these objects, called literals.

    The literals include the string, Unicode string, integer, float, long, list, tuple and dictionary types.

Q.3 **How many types of sequences are supported in Python?**

Ans: Three Types of Sequences are supported in python:

    (i)   String

    (ii)   List

    (iii)   Tuple

Q.4 **What factors guide the choice of identifiers in program?**

Ans: (i)   An identifier must start with a letter or underscore followed by any number of digits and/or letters.

    (ii)   No reserved word or standard identifier should be used.

    (iii)   No special character (Other than underscore) should be included in the identifier.

Q.5 **What is the difference between an expression and a statement in Python?**

Ans: A statement is an instruction that the Python interpreter can execute. We have only seen the assignment statement so far. Some other kinds of statements that we'll see shortly are while statements, forstatements, if statements, and import statements. (There are other kinds too!)

An expression is a combination of values, variables, operators, and calls to functions. Expressions need to be evaluated. If you ask Python to print an expression, the interpreter evaluates the expression and displays the result.

Q.6 **What are tokens in Python? How many types of tokens allowed in Python?**

Ans: Tokens are the smallest unit of the program. There are following tokens in Python:
- Reserved words or Keywords
- Identifiers
- Literals
- Operators
- Punctuators

Definition of all tokens may come. Which is not given in this question bank.

Q.7 **What are operators? What is their function? Give examples of some unary and binary operators.**

Ans: "Operators are those symbols used with operands, which tells compiler which operation is to be done on operands." in other words – —operators are tokens that trigger some computation/action when applied to variables and other objects in an expression."

Operators are of following types:

- **Unary operators** like (+) Unary Plus, (-) Unary Minus, not etc.
- **Binary Operators** like (+) addition, (*) multiplication, and etc.

Q.8 **What is block/code block/suit in Python?**

Ans: Sometimes a group of statements is part of another statement of function. Such a group of one or more statements is called **block** or **code-block** or **suit** in python. e.g.

```python
if a>b:
    print("A is graeter")
    print("Value of A is : ",a)
else:
    print("B is graeter")
    print("Value of A is : ",b)
```

Here both the sections are separate code-blocks

Q.9 **What is the role of indentation in Python?**

Ans: Indentation plays a very important role in Python. Python uses indentation to create blocks of code. Statements at same indentation level are part of same block/suit. You cannot unnecessarily indent a statement; python will raise an error for that.

Q.10 **How many types of strings are supported by Python?**

Ans: Python supports two types of strings:

(i)   Single-line string   That terminates in single line.

(ii)   Multi-line String   That stores multiple lines of text.

Skill Based Questions

Q.1 **What will be the sizes of following constants?**

(a) „\a"   (b) "\a"   (c) "Kumar\"s"   (d) „\""   (e) "it"s"

Ans: (a) 50   (b) 50   (c) 56   (d) 50   (e) 53

This screenshot is the output of the above question.

```
>>> sys.getsizeof(a)
28
>>> sys.getsizeof('\a')
50
>>> sys.getsizeof("\a")
50
>>> sys.getsizeof("kumar\'s")
56
>>> sys.getsizeof('\"')
50
>>> sys.getsizeof("it's")
53
```

Q.2 **How can you create multi-line strings in Python?**

Ans: We can create multi-line string by putting a backslash (\) at the end of line which allows you to continue typing in next line in same string.

```
>>> Text1="Hello\
World"
>>> Text1
'HelloWorld'
```

Q.3 **Which of the following are syntactically correct strings? State reason.**

(a) "Python is nice Language"

(b) „He called me "Friend!" when he came"

(c) "Very Good"

(d) „This is a good book"

(e) "Namaste

(f) "I liked „Harry Potter" very much"

Ans: (a) Correct   (b) Correct   (c) Incorrect (d) Correct   (e) Incorrect (f) Correct

Q.4 **What is the error in following Python program with one statement?**

print("My name is : ", name)

suggest a solution

Ans: Error is : —name 'name' is not defined‖ . And the solution is to declare the variable-name before this statement.

```
>>> name="aa"
>>> print("My name is : ", name)
My name is :  aa
```

Q.5 **Predict the output of the following:**

```
x,y=7,2
x,y,x=x+1,y+3,x+10
print(x,y)
```

Ans: Output: 17 5

Q.6 **What will be the output of the following code:**

```
name='Hari'
age=18
print(name,", you are ",age," now but ",end="")
print("You will be ",age+1," next Year")
```

Ans: Output: Hari , you are 18 now but You will be 19 next year

Q.7 **Write a Program to obtain temperature in Celsius and convert it into Fahrenheit using formula –**

C X 9/5 + 32 = F

```
c=int(input("Enter the value in Celsius"))
f =(c * 9/5) + 32
print("Temperature in fahrenheit is : ",f)
```

Ans:

Q.8 **Predict output:**

```
a,b,c=2,3,4
a,b,c=a*a,a*b,a*c
print(a,b,c)
```

Ans: Output: 4   6   8

Q.9 **WAP to read todays date (only date Part) from user. Then display how many days are left in the current month.**

Ans:
```
import datetime
td=0
now=datetime.datetime.now()
print(now.day)
if now.month==2:
     td=28
elif now.month in(1,3,5,7,8,10,12):
     td=31
else:
     td=30
print("Total remaining days in the current month are : ", td-now.day)
```

Q.10 **WAP to print the area of circle when radius of the circle is given by**
user. Ans:
```
r=int(input("Enter the radius : "))
area= 3.14*r*r
print("Area of circle is : ",area)
```

Q.11 **WAP to print the volume of a cylinder when radius and height of the cylinder is given by user.**

Ans:
```
r=int(input("Enter the radius : "))
h=int(input("Enter the height : "))
vol=3.14*r*r*h
print("Volume of Cylinder is : ",vol)
```

Q.12 **WAP that asks your height in centimeters and converts it into foot and inches.**

Ans:
```
cm=int(input("Enter height in Centimeters : "))
foot=cm//30
rcm=cm%30
inches=rcm*0.393701
print("Height is : ",foot," Foot ",inches," Inches ")
```

Q.13 **WAP to find area of a triangle.**

Ans:
```
import math
a=int(input("Enter side 1 on triangle : "))
b=int(input("Enter side 2 on triangle : "))
c=int(input("Enter side 3 on triangle : "))
s=(a+b+c)/2
area=s*math.sqrt((s-a)*(s-b)*(s-c))
print("Area of Triangle is : ",area)
```

Q.14 **WAP to calculate simple interest.**

Ans:
```
p=int(input("Enter Principal : "))
r=int(input("Enter Rates : "))
t=int(input("Enter Time :"))
si=(p*r*t)/100
print("Simple Interest is :",si)
```

Q.15 **WAP to read a number in n and prints n^2, n^3, n^4**

Ans:
```
n=int(input("Enter value of n : "))
print("n^2 : ",n*n)
print("n^3 : ",n*n*n)
print("n^4 : ",n*n*n*n)
```

Chapter – 3: DATA HANDLING

Very Short answer Type Questions

Q.1 **Identify the data types of the following values given bellow –**

3, 3j, 13.0, „12","14", 2+0j,19, [1,2,3],(3,4,5)

Ans: 3 – int   3j – complex   13.0 – float   ‗12' – string   ‒14‖ – string

2+0j – complex   19 – int   [1,2,3] – list   (3,4,5) – tuple

Q.2 **What will be the output of the following**

(a)12/4   (b)14//14   (c)14%4   (d) 14.0/4   (e) 14.0//4   (f)14.0%4

Ans: (a) 3.0   (b) 1   (c) 2   (d) 3.5   (e) 3.0   (f) 2.0

Q.3 **What will be the output of the following ?**
```
print(17//4)
print(17/4)
print(len(str(17//4)))
print(len(str(17/4)))
```

Ans: 4
4.25
1
4

Q.4 **What will be the output of the following ?**

(a) bool(0)   **(b) bool(„0")**   **(c) bool(int(„0"))**

(d) bool(str(0.0))   **(e) bool(0j)**   **(f) bool(0.0)**

Ans: (a) False   (b) True   (c) False

(d) True   (e) False   (f) False

Q.5 **What will be the output of the following ?**

(a)87//5   **(b)(87//5.0) == (87//5)**   **(c) 87//5.0**   **(d) 17%5.0**

Ans: (a) 17   (b) True   (c) 17.0   (d) 2.0

Q.6   **int(„a") produces error. Why?**

Ans: This is because _a' is an invalid literal for int() with base 10.

Q.7   **Write following expressions in Python.**

(a) _   **(b)**   √   **(c)** $\sqrt{}$ **(d)**

Ans: **(a)** (b*b*h)/3

(b)   d=math.sqrt(pow(x2-x1,2)+pow(y2-y1,2))

(c) x1=((-b) + math.sqrt((b*b)-(4*a*c)))/(2*a)

   x2=((-b) - math.sqrt((b*b)-(4*a*c)))/(2*a)

(d) pow(a,n) * pow(a,m) = pow(a,m+n)

Short Answer Type Questions

Q.1   **What are data types? What are Python"s built-in core data types?**

Ans: Every value in Python has a datatype. Since everything is an object in Python programming, data types are actually classes and variables are instance (object) of these classes.

There are various data types in Python. Some of the important types are listed below.

(i) Numbers   (ii) String   (iii) List   (iv) Tuple   (v) Dictionary

Q.2   **Which data types of Python handle Numbers?**

Ans: It is cleared by name that Number data types are used to store numeric value in Python. The Numbers in Python have following core data types:

(i)   Integers

a.   Integers (signed)

b.   Booleans

(ii)   Floating-Point Numbers

(iii)   Complex Numbers

Q.3   **Why is Boolean considered a subtype of Integers?**

Ans: Because Boolean Values *False* and *True* behave like the values 0 and 1, respectively. So Boolean type is a subtype of plain integers.

Q.4   **What do you understand by term „immutable"?**

Ans: Immutable types are those data types that can never change their value in place. In Python the following types are immutable:

(i)   integers

(ii)   floating-point numbers

(iii)   Booleans

(iv)   Strings

(v)   Tuples

Q.5   **What will be the output of the following code? Why?**

(a) 13 or len(13)   **(b) len(13) or 13**

Ans: (a) 13   (b) TypeError: object of type 'int' has no len().

Q.6   What are mutable and immutable types in Python? List both of them.

Ans: **Mutable types** means those data types whose values can be changed at the time of execution. They are as follows:

- Lists
- Dictionaries
- Sets

Immutable types are those data types that can never change their value in place. In Python the following types are immutable:

- integers
- floating-point numbers
- Booleans
- Strings
- Tuples

Q.7   What are augmented assignment operators? How are they useful?

Ans: An augmented assignment is generally used to replace a statement where an operator takes a variable as one of its arguments and then assigns the result back to the same variable. A simple example is x += 1 which is expanded to x = x + (1). Similar constructions are often available for various binary operators. They are helpful in making the source code small.

Skill Based Questions

Q.1   WAP to calculate compound simple interest after taking the principle, rate and time.

Ans:
```python
#Compund Interest
p=int(input("Enter the Principal"))
r=int(input("Enter the Interest Rate"))
t=int(input("Enter the Tenure"))
temp=1+r/100
f=1
for i in range(1,t+1):
    f=f*temp
Amount=p*f
interest=Amount-p
print("The interest on ",p," with rate ",r," is ",interest)
```

Q.2   WAP to check the given year is leap year or not.

Ans:
```python
year = int(input("Enter year"))
if (year % 4) == 0:
    if (year % 100) == 0:
        if (year % 400) == 0:
            print("{0} is a leap year".format(year))
        else:
            print("{0} is not a leap year".format(year))
    else:
        print("{0} is a leap year".format(year))
else:
    print("{0} is not a leap year".format(year))
```

Q.3 **WAP to take two numbers and check that the first number is fully divisible by second number or not.**

Ans:
```python
a=int(input("Enter First Number : "))
b=int(input("Enter Second Number : "))
if a%b==0:
    print(a," is fully divisible by ",b)
else:
    print(a," is not fully divisible by ",b)
```

Q.4 **What will be the output of the following?**
```python
a=5-4-3
b=3**2**3
print(a)
print(b)
```

Ans: -2

6561

Q.5 **What will be the output of the following?**
```python
x,y=4,8
z=x/y*y
print(z)
```

Ans: 4.0

Q.6 **WAP to take value of x,y,z from the user and calculate the equation**

Ans:
```python
x=int(input("Enter x : "))
y=int(input("Enter y : "))
z=int(input("Enter z : "))
f=4*pow(x,4)+3*pow(y,3)+9*pow(z,2)+6*3.14
print("The Answer is : ",f)
```

Q.7 **WAP to take the temperatures of all 7 days of the week and displays the average temperature of that week.**

Ans:
```python
d1=int(input("Temperature of day 1 : "))
d2=int(input("Temperature of day 2 : "))
d3=int(input("Temperature of day 3 : "))
d4=int(input("Temperature of day 4 : "))
d5=int(input("Temperature of day 5 : "))
d6=int(input("Temperature of day 6 : "))
d7=int(input("Temperature of day 7 : "))
avg=(d1+d2+d3+d4+d5+d6+d7)/7
print("The average temp is : ",avg)
```

Chapter – 4: CONDITIONAL AND ITERATIVE STATEMENTS

Short Answer Type Questions

Q.1 **What a range() function does? Give an example.**

Ans: The range() function returns a sequence of numbers, starting from 0 by default, and increments by 1 (by default), and ends at a specified number. its syntax is ***range(start, stop, step)*** e.g.

```python
x = range(3, 6)
for n in x:
    print(n)
#This code will print 3 4 5
```

```python
x = range(1, 10,2)
for n in x:
    print(n)
#This code will print 1 3 5 7 9
```

Q.2   What are loops in Python? How many types of loop are there in Python?

Ans: Loops are iteration constructs in Python. Iteration means repetition of a set of statements depending upon a condition test. Loops has three basic elements within it to repeat the statements –

- Initialization (Start)
- Check Condition (Stop)
- Updation (Step)

Python provide two types of loop

(i)   Conditional Loop **while**((Condition based loop)

(ii)   Counting loop **for** (loop for a given number of times).

Q.3   What is the syntax of if-elif statement in Python?

Ans: The syntax of if-elif statement in python is as follows:

> **If** condition1:
>> #code-block of statements when condition1 is true
>
> **elif** condion2:
>> #code-block of statements when condition2 is true
>
> **elif** condition3:
>> #code-block of statements when condition3 is true
>> .
>> .
>> .
>
> **else:**
>> #code-block of statements when all above conditions are false.

Q.4   What are jump statements in Python? Name jump statements with example.

Ans: Python offers two jump statements to be used with in loops to jump out of loop-iterations. These are ***break*** and ***continue*** statements.

```
#program to stop loop when 5 encounters
n=int(input("Enter limit : " ))
i=0
while i<=n:
    i+=1
    if i==5:
        break
    print(i)
```

```
#program to print upto n excluding 5
n=int(input("Enter limit : " ))
i=0
while i<=n:
    i+=1
    if i==5:
        continue
    print(i)
```

Q.5   Rewrite the following code fragment using for loop.

```
i=100
while(i>0):
    print(i)
    i-=3
```

Ans:
```
for i in range(100,0,-3):
    print(i)
```

Q.6   What is the error in following code. Rewrite the correct code.

```
weather='raining'
if weather='sunny':
    print('wear sunblock')
elif weather='snow':
    print('Going skiing')
else:
    print("None of the above")
```

Correct Code:
```
weather='raining'
if weather=='sunny':
    print('wear sunblock')
elif weather=='snow':
    print('Going skiing')
else:
    print("None of the above")
```

Q.7 **Rewrite the following code fragment using while loop.**

```
for i in range(1,10):
    if i%3==0:
        print(i)
```

Ans:
```
i=1
while i<10:
    if i%3 ==0:
        print(i)
    i+=1
```

Skill Based Questions

Q.1 **WAP that searches for prime numbers from 15 through 25.**

Ans:
```
for a in range(15,25):
    k=0
    for i in range(2,a//2+1):
        if(a%i==0):
            k=k+1
    if(k==0):
        print(a)
```

Q.2 **WAP to test if given number is prime or not.**

Ans:
```
a=int(input("Enter number: "))
k=0
for i in range(2,a//2+1):
    if(a%i==0):
        k=k+1
if(k<=0):
    print("Number is prime")
else:
    print("Number isn't prime")
```

Q.3 **WAP to compute the result when two numbers and one operator is given by user.**

Ans:
```
a=int(input("Enter 1st number: "))
b=int(input("Enter 2nd number: "))
c=input("Enter the Operation +,-,/,*:")
print("The result is : ", end='')
```

Q.4 **WAP to calculate the roots of a given quadratic equation.**

Ans:
```
import math
a=int(input("Enter a "))
b=int(input("Enter b "))
c=int(input("Enter c "))
d=(b*b)-(4*a*c)
if d>=0:
    print("roots are : ")
    x1=-b+math.sqrt(d)/(2*a)
    x2=-b-math.sqrt(d)/(2*a)
    print(" x1 = =",x1)
    print(" x2 = =",x2)
else:
    print("roots are imaginary.")
```

Q.5   **WAP to input a digit and print it in words.**

Ans:
```python
n=int(input("Enter the Digit from 0 to 9: "))
print("Entered Digit is : ",end='')
if n==0:
    print("Zero")
elif n==1:
    print("One")
elif n==2:
    print("Two")
elif n==3:
    print("Three")
elif n==4:
    print("Four")
elif n==5:
    print("Five")
elif n==6:
    print("Six")
elif n==7:
    print("Seven")
elif n==8:
    print("Eight")
elif n==9:
    print("Nine")
else:
    print("Not a digit")
```

Q.6   **WAP to check whether square root of a given number is prime or not.**

Ans:
```python
import math
n=int(input("Enter a number"))
m=int(math.sqrt(n))
k=0
for i in range(2,m//2+1):
    if(m%i==0):
        k=k+1
if(k==0):
    print(m,", which is sqare root of ",n,", is Prime number.")
else:
    print(m,", which is sqare root of ",n,", is not Prime number.")
```

Q.7   **WAP to print first n odd numbers in descending order.**

Ans:
```python
n=int(input("Enter the Limit "))
if n%2==0:
```

Q.8   **WAP to print the following series –**

(i)   1 4 7 10 40

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

Ans: (i)
```python
n=int(input("Enter the Limit "))
```

(ii)
```python
n=int(input("Enter the Limit "))
j=0
for i in range(1,n+1,3):
    print(pow(-1,j)*i, end='  ')
    j+=1
```

Q.9 **WAP to find the average of the list of the numbers entered through keyboard.**

Ans:
```
n=int(input("Enter the Limit "))
s=0
for i in range(1,n+1):
    print("Enter ",i,end='')
    a=int(input("th number : "))
    s=s+a
avg=s/n
print("The sum of entered numbers : ",s)
print("The Average of entered numbers : ",avg)
```

Q.10 **WAP to find the largest number from the list of the numbers entered through keyboard.**

Ans:
```
n=int(input("Enter the Limit less than 99999999999 "))
small=99999999999
for i in range(1,n+1):
```

Q.11 **WAP to find the 2nd largest number from the list of the numbers entered through keyboard. (This program is from List Chapter)**

Ans:
```
a=[]
n=int(input("Enter number of elements:"))
for i in range(1,n+1):
    b=int(input("Enter element:"))
    a.append(b)
a.sort()
print("Second largest element is:",a[n-2])
```

Q.12 **WAP to find the sum of n natural numbers.**

Ans:
```
n=int(input("Enter the Limit : "))
s=0
for i in range(1,n+1):
    s=s+i
print("The sum is : ",s)
```

Q.13 **WAP to find the sum of first n even numbers.**

Ans:
```
n=int(input("Enter the Limit : "))
s=0
for i in range(0,n+1,2):
    s=s+i
print("The sum is : ",s)
```

Q.14 **WAP to find the sum of first n odd numbers.**

Ans:
```
n=int(input("Enter the Limit : "))
s=0
for i in range(1,n+1,2):
    s=s+i
print("The sum is : ",s)
```

Q.15 **WAP to print the following pattern**

	(a)		(b)		(c)		(d)	
	*		*		A		0	
	* *		* *		A B		2 2	
	* * *		* * *		A B C		4 4 4	
	* * * *		* * * *		A B C D		8 8 8 8	
	* * * * *		* * * * *		A B C D E			

Ans: (a)
```python
n=int(input("Enter the Limit : "))
for i in range(1,n+1):
    for j in range(1,i+1):
        print("*",end='')
    print("")
```

(b)
```python
n=int(input("Enter the Limit : "))
for i in range(1,n+1):
    for k in range(n-i,0,-1):
        print(' ',end='')
    for j in range(1,i+1):
        print("*",end='')
    print("")
```

(c)
```python
s=input("Enter the String : ")
n=len(s)
for i in range(0,n):
    for j in range(0,i):
        print(s[j],end='')
    print("")
```

(d)
```python
n=int(input("Enter the Limit : "))
for i in range(0,n+1,2):
    for j in range(0,i+1,2):
        print(i,end='')
    print("")
```

Chapter – 6: DEBUGGING PROGRAMS

Short answer Type Questions

Q.1 **What do you understand by Syntax errors and Semantics errors?**

Ans: **Syntax Errors:** syntax error occur when rules of a programming language are misused i.e. grammatical rule of Python is violated. e.g.

X<-x*y

if x=(x*y) etc.

Semantics Errors: Semantics error occur when statements are not meaningful. e.g.

x * y = z

this will result in a semantical eoor as an expression cannot come on the left side of an assignment operator.

Q.2   Why are logical errors harder to locate?

Ans: in spite of logical errors presence, program executes without any problems but the output produced is not correct. Therefore, each and every statement of the program needs to be scanned and interpreted. Thus the logical errors are harder to locate.

Q.3   What is an Exception?

Ans: Exception in general refers to some contradictory or unusual situation which can be encountered unexpectedly while executing the program. Unhandled exceptions will cause Python to halt execution.

Q.4   Why is Exception Handling is required?

Ans: Unhandled exceptions will cause Python to halt execution. The exception handling is ideal for processing exceptional situations in a controlled way so that program ends gracefully rather than abrupt crashing of the program.

Q.5   What is the need for debugger tool?

Ans: Debugger tools are very useful especially if the code is big or the error is not very clear, it becomes very difficult to manually figure out the origin and cause of the problem. Debugger tools here prove very handy and useful. They show us the line by line execution and its result on variables interactively and help a programmer get to the root of the problem.

Q.6   What are main error types? Which types are most dangerous and why?

Ans: Main error types are -

(i)Compile-time errors   (ii) Run-time errors   (iii) Logical errors

Logical errors are most dangerous errors because these are most difficult to fix. The error is caused by a mistake in the program's logic. You won't get an error message, because no syntax or runtime error has occurred. You will have to find the problem on your own by reviewing all the relevant parts of your code – although some tools can flag suspicious code which looks like it could cause unexpected behaviour.

Q.7   What is a difference between an error and exception?

Ans: Exception and Error: Exceptions are those which can be handled at the run time whereas errors cannot be handled. An exception is an Object of a type deriving from the System. Exception class. System Exception is thrown by the CLR (Common Language Runtime) when errors occur that are nonfatal and recoverable by user.

Q.8   Name some common built-in exceptions in Python.

Ans: Some Common built-in Exceptions in Python are:

(i) EOFError   (ii) IOError   (iii) NameError   (iv) IndexError   (v) ImportError

(vi) TypeError   (vii) ValueError   (viii) ZeroDivisionError   (ix) KeyError

Q.9   when does these exception occur?

(a) Type Error   (b) Index Error   (c) Name Error

Ans: (a) Type Error: Raised when an operation or function is applied to an object of inappropriate type. e.g. if you try to compute square-root of a string.

(b) Index Error: Raised when a sequence subscript or index is out of range e.g. from a string of length 4 if you try to read a value of index 4 or more.

(c) Name Error: Raised when an identifier name is not found.

Q.10 **What is debugging and code tracing?**

Ans: Debugging involves correction of code so that the cause of errors is removed. I other words we can say that debugging means figure out the origin of error in code, fix the error code and review and rerun your code to ensure that the error is fixed.

Chapter – 12: COMPUTER SYSTEM OVERVIEW

Very Short answer Type Questions

Q.1 **What is volatile memory?**

Ans: RAM is known as Volatile Memory because when we switch off the computer its data is vanished.

Q.2 **Define each of the following:**

(a) byte   (b) kilobyte (c) megabyte   (d) gigabyte   (e) terabyte

Ans: (a) byte: This is the unit of memory in computer. 1 byte = 8 bits

(b) kilobyte: This is the unit of memory in computer. 1 kilobyte = 1024 bytes

(c) megabyte: This is the unit of memory in computer. 1 megabyte = 1024 kilobytes

(d) gigabyte: This is the unit of memory in computer. 1 gigabyte = 1024 megabytes

(e) terabyte: This is the unit of memory in computer. 1 terabyte = 1024 gigabytes.

Short Answer Type Questions

Q.1 **State the basic units of computer. Name the subunits that make up the CPU, and give the function of each of the unit.**

Ans: Basic units of computer are Input Unit, Central Processing Unit and Output Unit. Sub unites of CPU are Arithmetical Logical Unit(ALU), Control Unit (CU) and Memory Unit(MU).

Q.2 **What is the function of memory? What are its measuring units?**

Ans: The computer memory is a temporary storage area. It holds the data and instructions that the Central Processing Unit (CPU) needs. Before a program can run, the program is loaded from some storage medium into the memory. This allows the CPU direct access to the program. Its measuring units are byte, kilobyte, megabyte, gigabyte, terabyte etc.

Q.3 **Why is primary memory termed as „destructive write" memory but „non-destructive read" memory?**

Ans: The primary memory is called destructive write because the data enter here are temporary. That's why your RAM gets cleared after every restart.

Q.4 **What is the role of CPU of a mobile system?**

Ans: A mobile processor is found in mobile computers and cellphones.
A CPU chip is designed for portable computers, it is typically housed in a smaller chip package, but more importantly, in order to run cooler, it uses lower voltages than its desktop counterpart and has more sleep mode capability. A mobile processor can be throttled down to different power levels or sections of the chip can be turned off entirely when not in use. Further, the clock frequency may be stepped down under low processor loads. This stepping down conserves power and prolongs battery life.

Q.5 **What is SoC? how it is different from CPU? Why is it considered a better development?**

Ans: A system on a chip (SoC) combines the required electronic circuits of various computer components onto a single, integrated chip (IC). SoC is a complete electronic substrate system that may contain analog, digital, mixed-signal or radio frequency functions. Its components usually include a graphical processing unit (GPU), a central processing unit (CPU) that may be multi-core, and system memory (RAM).

Because SOC includes both the hardware and software, it uses less power, has better performance, requires less space and is more reliable than multi-chip systems. Most system-on-chips today come inside mobile devices like smartphones and tablets. These are considered a better development because of their small size and speed capability.

Q.6 **What are various categories of software?**

Ans: Software are classified into following categories –

(i) System Software

    a. Operating System

    b. Language Processor

(ii) Application Software

    a. Packages

    b. Utilities

    c. Customized software

    d. Developer Tools

Q.7 **What is the difference between an interpreter and a compiler?**

Ans: Interpreter: Interpreter is a type of system software that translates and executes instructions written in a computer program lini-by-line, unit by unit etc. It is slower in execution because each time when you run the program translation is required.

Compiler: Compiler is another type of system software that translates and executes instructions written in a computer program in one go. Once compiled program need not to translate again so it works faster.

Q.8   What is application software? Why it is required?

Ans: Application software is the set of programs necessary to carry out operations for a specific task. Such as for word processing there are many application software like MS-Word, Wordpad etc. These software are required to perform special task using the computer like painting, recording, typing, data handling etc.
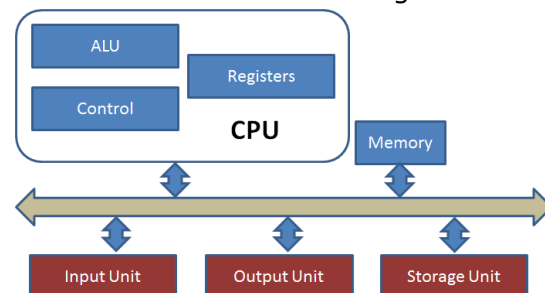
Q.9   Briefly explain the basic architecture of a computer.

Ans: Computer organization refers to logical structure of a computer describing how its components are connected to one another, how they affect one another's functioning and contributes to overall performance of computer.

Computers follow the _IPO' principal i.e.

Input →Process →Output

(That means a certain input is processed to Generate specific output)

Q.10   What do you understand by input unit? What is its significance? What does computer system consist of?

Ans: Input unit is formed by the input devices(Keyboard, mouse, MICR, OBCR etc.) attached to the computer. Input unit is responsible for taking input and converting it into computer understandable form(the binary code). Some common input devices are:

(i)   Keyboard

(ii)   Mouse

(iii)   Microphone

(iv)   Scanner

(v)   Webcam

(vi)   Optical Bar Code Reader

(vii)   Optical Mark Reader

(viii)   Magnetic Ink Character Reader

(ix)   Touch Pad

(x)   Track Ball

(xi)   Joy stick

(xii)   Touch Screen

(xiii)   Biometric Sensors. Etc.

Q.11   What is the function of CPU in a computer system? What are its sub units?

Ans: The central processing unit (CPU) of a computer is a piece of hardware that carries out the instructions of a computer program. It performs the basic arithmetical, logical, and input/output operations of a computer system. The CPU is like the brains of the computer - every instruction, no matter how simple, has to go through the CPU. So let's say you press the letter 'k' on your keyboard and it appears on the screen - the CPU of your computer is what makes this possible. The CPU is sometimes also referred to as the central processor unit, or processor for short. So when you are looking at the specifications of a computer at your local electronics store, it typically refers to the CPU as the processor. Its sub units are:

(i) Control Unit   (ii) Arithmetical and Logical Unit (ALU)   (iii) Memory Unit

Q.12   What functions are performed by the control unit?

Ans: The CU controls and guides the interpretation, flow and manipulation of all data and information. CU sends control signals until the required operations are done properly by ALU and memory. Another function of CU is the program execution. It means carrying out all the instructions stored in the program. CU gets program instruction from the memory and executes them one after the other. CU acts as a supervisor by controlling and guiding the operation taking place.

Q.13   Distinguish between CPU and ALU?

Ans: Difference Between **ALU and CPU** is that arithmetic logic unit (ALU), another component of the processor, performs arithmetic, comparison, and other operations. While Processor also central processing unit (CPU), interprets and carries out the basic instructions that operate a computer.

The main difference between CPU and ALU is that the CPU is an electronic circuit that handles instructions to operate the computer while the ALU is a subsystem of the CPU that performs arithmetic and logical operations.

Q.14   What is the function of output unit in a computer system?

Ans: Input devices are the hardware that give computers instructions. Output devices relay the response from the computer in the form of a visual response (monitor), sound (speakers) or media devices (CD or DVD drives). The purpose of these devices is to translate the machine's response to a usable form for the computer user.

Q.15   Distinguish between internal and external memory.

Ans: Internal memory is usually chips or modules that you attach directly to the motherboard. Internal Memory is a circular disc that continuously rotates as the computer accesses its data. External memory often comes in the form of USB flash drives; CD, DVD, and other optical discs; and portable hard drives.

Q.16   What are RAM and ROM? How are they alike? How are they different? What are PROM, EPROM, EEPROM?

Ans: A ROM chip is a non-volatile storage medium, which means it does not require a constant source of power to retain the information stored on it.

A RAM chip is volatile, which means it loses any information it is holding when the power is turned off.

Both of them are known as primary memory as they can directly work with CPU.

Read Only Memory (ROM)

Programmable Read Only Memory (PROM)

Erasable Programmable Read Only Memory (EPROM)

Electrically Erasable Programmable Read Only Memory (EEPROM)

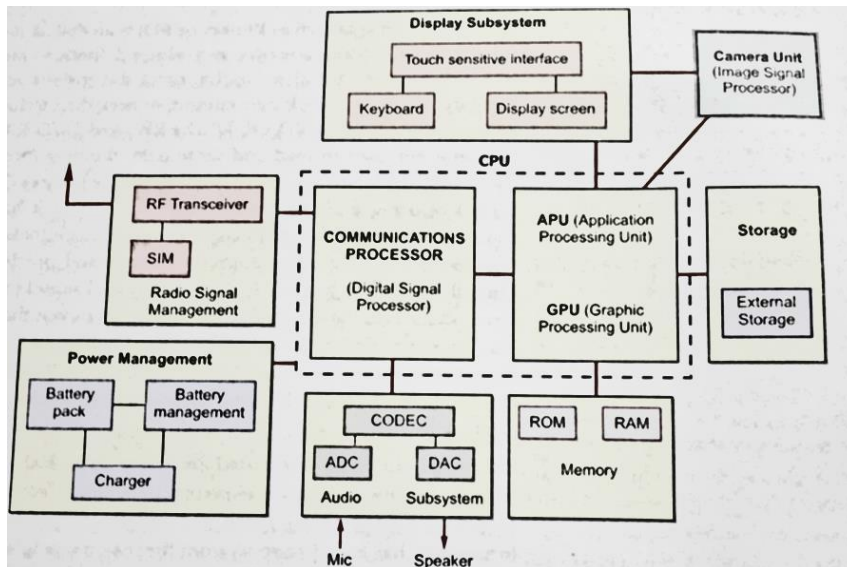**Q.17   What are major functional components of a mobile system?**

Ans: Major functional components of a mobile system are-

(i)   Mobile Processor

   a. Communications Processing Unit

   b. Application Processing Unit

   c. GPU (Graphics Processing Unit)

(ii)   SoC (System on a chip)

(iii)   Display Subsystem

a. Display Screen

b. Touch Sensitive Interface

c. Touch Sensitive Keyboards

(iv)   Camera Subsystem

(v)   Mobile System Memory

a. RAM

b. ROM

(vi)   Storage

(vii)   Power Management Subsystem

Q.18 Draw a block diagram depicting organization of a mobile system.

Ans:

Q.19 What is the role of Power Management Unit in a mobile system?

Ans: This subsystem is responsible for providing power to a mobile system. The mobile systems work on limited power provided through an attached battery unit. This system has a battery management system that works with a battery charger and a battery unit and provides power to the mobile system in required form.

It also contains a collection of different functions like battery charging, monitoring and supplying many different voltages these systems require. It also contains software controlled turn-on and turn-off feature to optimize the power consumption and battery life.

Q.20 What does the communication processor do?

Ans: this subsystem is responsible for making and receiving phone calls on a mobile handset. It has a digital signal processor that helps it work with RF Transceiver and the Audio subsystem.

Q.21 What is GPU? How is it useful?

Ans: Graphics Processing Unit assists the CPU by handling the visuals, other graphically-rich applications. In short, GPU handles all graphics-related chores of a mobile CPU.

Q.22 What is system software?

Ans: The software that controls internal computer operations is called system software. It manages all the resources of a system. Its example is Operating System.

Q.23 Discuss the role of utility software in the context of computer performance?

Ans: Utilities are those application programs that assist the computer by performing housekeeping functions like backing up disk or scanning/cleaning viruses or arranging information etc. its example is Antivirus software.

Q.24 What is the importance of an OS?

Ans: An operating system is a program which acts as an interface between a user and the hardware. It manages all the resources of the computer system. It provides and environment to the user to work with. All the application are installed after the operating system is installed. It manages memory, processing, storage, memory etc.

Q.25 How are software libraries useful? Name some software libraries of Python.

Ans: A software library is a predefined and available to use, suit of data and programming code in the form of prewritten code/ functions/scripts/classes etc. that can be used in the development of the new software programs and applications.

Some software library in python are:

(i)   NumPy (numerical Python)

(ii)   SciPy (Scientific Python)

(iii)   Pandas Library

Q.26 What is the utility of these software?

(a) disk fragmentor   (b) backup software

Ans: (a) disk fragmentor: A file is fragmented when it becomes too large for your computer to store in a single location on a disk. When this happens, your computer splits the file up and stores in pieces. You can use fragmented files, but it takes your computer longer to access them.

(b) Backup software: This utility program facilitates the backing-up of disk. Back-up means duplicating the disk information so that in case of any damage or data-loss, this backed-up data may be used.

Chapter – 13: DATA REPRESENTATION

Very Short answer Type Questions

Q.1 Add the binary numbers (a) 110101 and 101111   (b) 10110 and 1101

Ans: (a) 1100100   (b) 100011

Q.2 Convert 11111011110101_2 to octal.

Ans: 37365

Q.3 Covert the following binary numbers to decimal -

(a)1010   (b) 111000   (c) 10101111   (d) 10110

Ans: (a) 10   (b) 56   (c) 175   (d) 22

Q.4 Covert the following Decimal numbers to binary -

(a) 23   (b) 100   (c) 161   (d) 145

Ans: (a) 10111   (b) 1100100   (c) 10100001   (d) 10010001

Q.5 Covert the following Hexadecimal numbers to Binary -

(a) BE   (b) BC9   (c) A07   (d) 7AB4

Ans: (a)10111110   (b) 101111001001   (c) 101000000111   (d) 0111101010110100

Q.6 Covert the following binary numbers to Hexadecimal -

(a)101000001   (b) 11100011   (c) 10101111   (d) 101101111

Ans: (a) 141   (b) E3   (c) AF   (d)16F

Q.7 **Covert the following Octal numbers to Binary -**

(a) 456 **(b) 26** **(c) 751** **(d) 777**

Ans: (a) 100101110 (b) 010110 (c) 111101001 (d)111111111

Q.8 **Convert the following:**

(a) 446_8 to (**$)_{16}$** **(b) 47.5_8 to (** **$)_{10}$** **(c) 45.9_{10} to (** **$)_2$**

Ans: (a) 126 (b) 39.625 (c) 101101.1110

Short Answer Type Questions

Q.1 **What is the use of encoding schemes?**

Ans: A character encoding provides a key to unlock (ie. crack) the code. It is a set of mappings between the bytes in the computer and the characters in the character set. Without the key, the data looks like garbage.

So, when you input text using a keyboard or in some other way, the character encoding maps characters you choose to specific bytes in computer memory, and then to display the text it reads the bytes back into characters. Unfortunately, there are many different character sets and character encodings, ie. many different ways of mapping between bytes, code points and characters. The section Additional information provides a little more detail for those who are interested.

Q.2 **Discuss UTF-8 encoding Scheme.**

Ans: UTF-8 is a compromise character encoding that can be as compact as ASCII (if the file is just plain English text) but can also contain any unicode characters (with some increase in file size). UTF stands for Unicode Transformation Format. The '8' means it uses 8-bit blocks to represent a character.

Q.3 **How UTF-8 encoding scheme different from UTF-32 encoding scheme?**

Ans: **UTF-8:** Variable-width encoding, backwards compatible with ASCII. ASCII characters (U+0000 to U+007F) take 1 byte, code points U+0080 to U+07FF take 2 bytes, code points U+0800 to U+FFFF take 3 bytes, code points U+10000 to U+10FFFF take 4 bytes. Good for English text, not so good for Asian text.

UTF-32 uses 32-bit values for each character. That allows them to use a fixed-width code for every character. UTF-32 is opposite, it uses the most memory (each character is a fixed 4 bytes wide), but on the other hand, you *know* that every character has this precise length, so string manipulation becomes far simpler. You can compute the number of characters in a string simply from the length in bytes of the string. You can't do that with UTF-8.

Q.4 **What are ASCII and extended ASCII schemes?**

Ans: The standard ASCII character set uses just 7 bits for each character. There are several larger character sets that use 8 bits, which gives them 128 additional characters. The extra characters are used to represent non-English characters, graphics symbols, and mathematical symbols.

The extended ASCII character set uses 8 bits, which gives it an additional 128 characters. The extra characters represent characters from foreign languages and special symbols for drawing pictures.

Q.5 **What is the utility of ISCII encoding schemes?**

Ans: ISCII is a bilingual character encoding (not glyphs) scheme. Roman characters and punctuation marks as defined in the standard lower-ASCII take up the first half the character set (first 128 slots). Characters for indie languages are allocated to the upper slots (128-255). T

Q.6   What is Unicode? What is its significance?

Ans: Unicode is a character encoding standard that has widespread acceptance. Microsoft software uses Unicode at its core. Whether you realize it or not, you are using Unicode already! Basically, ―computers just deal with numbers. They store letters and other characters by assigning a number for each one. Before Unicode was invented, there were hundreds of different encoding systems for assigning these numbers. No single encoding could contain enough characters.1‖ This has been the problem we, in SIL, have often run into. If you are using a legacy encoding your font conflicts with the font someone in another area of the world uses. You might have an in your font while someplace else someone used a at the same codepoint. Your files are incompatible. Unicode provides a unique number for every character and so you do not have this problem if you use Unicode. If your document calls for U+0289 it will be clear to any computer program what the character should be

Q.7   What are ASCII and ISCII? Why are these used?

Ans: ASCII uses a 7-bit encoding and ISCII uses an 8-bit which is an extension of ASCII. These are encoding schemes to represent character set in s computer system.

Q.8   Compare UTF-8 and UTF-32 encoding schemes. Which one is most popular scheme?

Ans: UTF-8: Variable-width encoding, backwards compatible with ASCII. ASCII characters (U+0000 to U+007F) take 1 byte, code points U+0080 to U+07FF take 2 bytes, code points U+0800 to U+FFFF take 3 bytes, code points U+10000 to U+10FFFF take 4 bytes. Good for English text, not so good for Asian text.

UTF-32 uses 32-bit values for each character. That allows them to use a fixed-width code for every character. UTF-32 is opposite, it uses the most memory (each character is a fixed 4 bytes wide), but on the other hand, you *know* that every character has this precise length, so string manipulation becomes far simpler. You can compute the number of characters in a string simply from the length in bytes of the string. You can't do that with UTF-8.

Q.9   What do you understand by code point and code unit?

Ans: A code point is the atomic unit of information. ... Each code point is a number which is given meaning by the Unicode standard. A code unit is the unit of storage of a part of an encoded code point. In UTF-8 this means 8-bits, in UTF-16 this means 16-bits.

Q.10   What is code space? How is it related to code point?

Ans: In computing, Code space may refer to: In memory address space: code space, where machine code is stored. For a character encoding: code space (or code space), the range of code points.

Chapter – 15: INSIGHT INTO PROGRAM EXECUTION

Short Answer Type Questions

Q.1   What are the front end and back end phases of compiler?

Ans: The front end phase of compiler is **Analysis Phase**. This phase of compilation identifies all the tokens in the source code and creates a symbol table with it.

The back end phase of compiler is **Synthesis Phase.** This phase of compiler parses the code and generates syntax tree, analyzing the syntax of the source code.

Q.2   Differentiate between compiler and interpreter.

Ans:

Aspects	Compiler	Interpreter
Input	A Compiler takes an entire program as its input.	It takes a single line of code or single instruction(such as loop) as its input.
Output	It generates intermediate object code from the whole program.	It does not create any intermediate object code.
Memory	Requires more memory during compilation.	Requires less memory during interpretation as interprets single instruction/unit-of-code at a time.
Errors	Displays lists of errors of entire program with line numbers and the error(s) in that line.	Displays the error of single line it is interpreting. Thus errors also appear one line at a time.
Always Required	No. once the role of compiler is over, a compiler is no longer required to run the executable fuile.	Always required. Interpreter translates and runs one instruction every time. So it is always required in memory to run the code.
Workload	Compilation once done, stays always. Recompilation not required if the code(without any change) needs to run again. The same executable file can be rerun without the compiler.	Interpretation is required each time you have to run the code. The interpreter does not create an executable and hence it is required every time, the code is to run.

Q.3   What are the characteristics of

compiler? Ans:

Aspects	Compiler
Input	A Compiler takes an entire program as its input.
Output	It generates intermediate object code from the whole program.
Memory	Requires more memory during compilation.
Errors	Displays lists of errors of entire program with line numbers and the error(s) in that line.
Always Required	No. once the role of compiler is over, a compiler is no longer required to run the executable fuile.
Workload	Compilation once done, stays always. Recompilation not required if the code(without any change) needs to run again. The same executable file can be rerun without the compiler.

Q.4   What are the characteristics of

Interpreter? Ans:

Aspects	Interpreter
Input	It takes a single line of code or single instruction(such as loop) as its input.
Output	It does not create any intermediate object code.
Memory	Requires less memory during interpretation as interprets single instruction/unit-of-code at a time.
Errors	Displays the error of single line it is interpreting. Thus errors also appear one line at a time.
Always Required	Always required. Interpreter translates and runs one instruction every time. So it is always required in memory to run the code.
Workload	Interpretation is required each time you have to run the code. The interpreter does not create an executable and hence it is required every time, the code is to run.