1. C) %
2. B) 0
3. C) 24
4. A) 2
5. D) 6
6. C) the finally block will be executed no matter if the try block raises an error or not.
7. A) It is used to raise an exception.
8. C) in defining a generator
9. A) _abc
10. D) all of the above

11.
```
def factorial(x):

  if x == 1:

    return 1

  else:

    return (x * factorial(x-1))

  num=input(printf("enter the num whose factorial is to be fund))

result = factorial(num)

print("The factorial of", num, "is", result)
```

12.
```
num = int(input("Enter any number : "))
if num > 1:
  for i in range(2, num):
    if (num % i) == 0:
      print(num, "is NOT a prime number")
      break
  else:
    print(num, "is a PRIME number")
elif num == 0 or 1:
  print(num, "is a neither prime NOR composite number")
else:
  print(num, "is NOT a prime number it is a COMPOSITE number")
```

13.
```
my_str = 'aIbohPhoBiA'

my_str = my_str.casefold()

rev_str = reversed(my_str)

# check if the string is equal to its reverse
if list(my_str) == list(rev_str):
  print("The string is a palindrome.")
else:
  print("The string is not a palindrome.")
```

14.
```
def pythagoras(opposite_side,adjacent_side,hypotenuse):
  if opposite_side == str("x"):
    return ("Opposite = " + str(((hypotenuse**2) - (adjacent_side**2))**0.5))
  elif adjacent_side == str("x"):
    return ("Adjacent = " + str(((hypotenuse**2) - (opposite_side**2))**0.5))
```

```python
        elif hypotenuse == str("x"):
            return ("Hypotenuse = " + str(((opposite_side**2) + (adjacent_side**2))**0.5))
        else:
            return "You know the answer!"

    print(pythagoras(3,4,'x'))
    print(pythagoras(3,'x',5))
    print(pythagoras('x',4,5))
    print(pythagoras(3,4,5))
```

15.
```python
string=input("Enter the string: ")
char=input("Please enter the char to find frequency of ta character\n")
count=0
for i in range(len(string)):
if(string[i]==char):
count=count+1
print("The frequency of the ",char,"in the string is: ",count)
```