QB365 Question Bank Software Study Materials

Control Structures Important 2, 3 & 5 Marks Questions With Answers (Book Back and Creative)

12th Standard

Computer Science

Total Marks : 75

2 Marks

10 x 2 = 20

1) List the control structures in Python.

Answer : There are three important control Structures

(i) Sequential

(ii) Alternative or Branching

(iii) Iterative or Looping

2) Write note on break statement.

Answer : The break statement terminates the loop containing it. Control of the program flows to the statement immediately after the body of the loop.

3) Write is the syntax of if..else statement.

Answer : **Syntax:**

```
if < condition > :
        statements-block 1
else:
        statements-block 2
```

4) Define control structure.

Answer : A program statement that causes a jump of control from one part of the program to another is called control structure or control statement.

5) Write note on range () in loop.

Answer : The range() is a built-in function, to generate series of values between two numeric intervals.

The syntax of range() is as follows:

range (start,stop,[step])

Where,

start - refers to the initial value

stop - refers to the final value

step - refers to increment value, this is optional part.

6) Write the syntax of for loop.

Answer : **Syntax:**

```
for counter_variable in sequence:
statements-block 1
[else: # optional block
statement-block 2]
```

7) What are the values taken by range ()?

Answer : It take values from string, list, dictionary.

8) What is sequential statement?

Answer : A sequential statement is composed of a sequence of statements which are executed one after another.

9) Write a program to print single-digit even numbers.

Answer : Program to illustrate the use of for loop- to print single digit even number.

```
for i in range (2,10,2):
Print (i, end = ' ')
```

Output:

```
2468
```

10) Write a program to display all 3 digit even numbers.

Answer :

```
for i in range (100 , 1000, 2):
print  (i, end = '\t')
else:
print ("\n End of the loop")
```

<u>**3 Marks**</u> 10 x 3 = 30

11) Write a program to display

A

A B

A B C

A B C D

A B C D E

Answer :

```
for i in range (1, 6) :
for j in range (65, 65 + i)
         a = chr(j)
print a
print
```

12) Write note on if..else structure.

Answer : (i) The if.. else statement provides control to check the true block as well as the false block,

(ii) Syntax :

```
    if < condition > :
         statemets-block 1
    else :
         statement-block 2
```

(iii) if..else statement thus provides possibilities and the condition determines which BLOCK is to be executed.

13) Using if..else..elif statement write a suitable programs to display largest of 3 numbers.

Answer : **Code :**

```
n1 = int(input(:Enter the first number:"))
n2 = int(input("Enter the second number:")
n3 = int(input(:Enter the third number:")
if(n1?=n2) and(n1 > =n3) :
        biggest = n1 ;
elif(n2 > =nl)and (n2 > =n3) :
        biggest = n2
else:
        biggest = n3
print("The biggest number between",n1,",",n2"and"n3"is",biggest)
```

Output :

```
Enter the first number : 1
Enter the second number : 3
Enter the third number : 5
The biggest number between 1, 3 and 5 is 5
```

14) Write the syntax of while loop.

Answer : **Syntax :**

```
while < condition > ;
          statements block 1
 [else:
          statements block2]
```

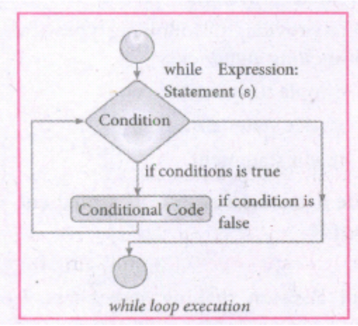
15) List the differences between break and continue statements.

Answer :

Break	**Continue**
The break statement terminates the loop containing it.	The continue statement is used to skip the remaining part of a loop.
Control of the program flows to the statement immediately after the body ot the loop.	Control of the program flows start with next iteration
Syntax : break	Syntax : continue

16) Draw a flowchart that illustrates the working of while loop.

Answer :



while loop execution

17) What will be the range of values displayed by the following?

Answer : range (1,30,1) - will start the range of values from 1 and end at 29

range (2,30,2) - will start the range of values from 2 and end at 28

range (30,3,-3) - will start the range of values from 30 and end at 6

range (20) - will consider this value 20 as the end value(or upper limit) and starts the range count from 0 to 19 (remember always ranget) will work till stop -1 value only)

18) Why we need to construct the pass statement?

Answer : pass statement is generally used as a placeholder. When we have a loop or function that is to be implemented in the future and not now, we cannot develop such functions or loops with empty body segment because the interpreter would raise an error. So, to avoid this we can use pass statement to construct a body that does nothing.

19) Write a Program to check the number is odd or even.

Answer :

```
# Program to check if the accepted number odd or even
a = int(input ("Enter any number:")
if a % 2 = = 0 :
print (a, "is an even number")
else:
print (a, "is an odd number")
```

Output 1:

```
Enter any number: 56
56 is an even number
```

Output 2:

```
Enter any number: 67
67 is an odd number.
```

20) Write a program to illustrate the use of string in range ( ).

Answer : # Programs to illustrate the use of string in range ( ) of for loop.

```
for word in 'computer':
print (word, end = ' ')
else:
print ("\n End of the loop")
```

Output:

```
c o m p u t e r
End of the loop
```

5 Marks 5 x 5 = 25

21) Write a detail note on for loop.

Answer : (i) **for loop** : The for loop usually known as a definite loop, because the programmer knows exactly how many times the loop will be executed.

(ii) **Syntax:**

```
for counter_variable in sequence:
    statements - block 1
[else: # optional block
    statements - block 2]
```

(iii) The for ... in statement is a looping statement used in Python to iterate over a sequence of objects, i.e., it goes through each item in a sequence. Here the sequence is the collection of ordered or unordered values or even a string.

(iv) **range ():** The range() is a built-in function,to generate series of values between two numeric intervals

(iv) The syntax of range() follows :

range (start, stop, [step])

Where,

start - refers to the initial value

stop - refers to the final values

step - refers to increment value, this is optional part.

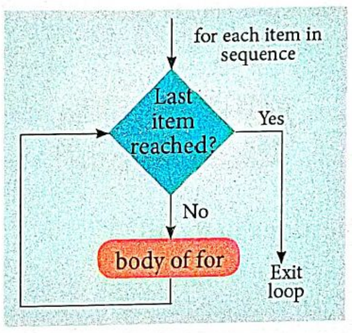
Examples for range( ):

range (1,30,1) - will start the range of values from 1 and end at 29

range (2,30,2) - will start the range of values from 2 and end at 28

range (30,3,-3) - will start the range of values from 30 and end at 6.

range (20) - will consider this value 20 as the end value and starts the range count from 0 to 19



Example :

```
# Program to illustrate the use of for loop - to print single digit even number
for i in range (2,10,2):
    print (i, end=' ')
```

Output :

```
2 4 6 8
```

22) Write a detail note on if..else..elif statement with suitable example.

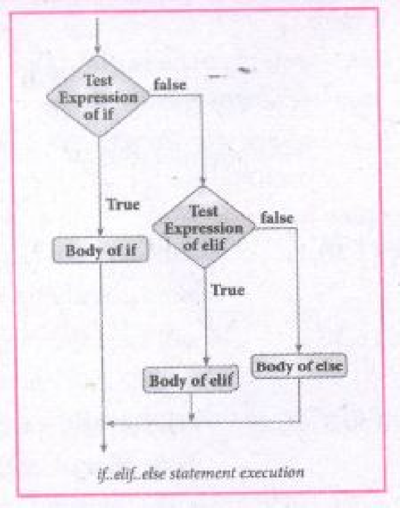
Answer : **Nested if..elif..else statement :**

(i) When we need to construct a chain of **if** statement(s) then '**elif**' clause can be used instead of '**if else**'.

(ii) **Syntax:**

```
if < condition - 1 > :
      statements-block 1
elif < condition - 2 > :
       statements-block 2
else:
       statements-block n
```

(iii) In the syntax of **if..elif ..else** mentioned above, condition -1 is tested if it is true then statements-block 1 is executed, otherwise, the control checks condition-2, if it is true statements-block 2 is executed and even if it fails statements-block n mentioned in **else** part is executed.



if..elif..else statement execution

(iv) '**elif**' clause combines **if..else-if ..else** statements to one **if..elif … else**. '**elif**' can be considered to be abbreviation of '**else if**'. In an '**if**' statement there is no limit of 'elif' clause that can be used, but an '**else**' clause if used should be placed at the end.

(v) **Example:** # Program to illustrate the use of nested if statement

Average	Grade
> = 80 and above	A
> = 70 and < 80	B
> = 60and < 70	C
> = 50 and < 60	D
Otherwise	E

```
 m1 = int (input("Enter mark in first subject:"))
m2 = int (input("Enter mark in second subject:"))
avg = (m1+m2)/2
if avg > =80:
   print ("Grade : A")
elif avg > =70 and avg < 80:
   print ("Grade : B")
elif avg > =60 and avg < 70:
   print ("Grade : C")
elif avg > =50 and avg < 60:
   print ("Gradec: D")
else:
   print("Grade : E")
```

Output 1:

```
Enter mark in first subject : 34
Enter mark in second subject : 78
Grade : D
```

Output 2:

```
Enter mark in first Subject : 67
Enter mark in  second subject : 73
Grade: B
```

23) Write a program to display all 3 digit odd numbers.

Answer : for i in range (100, 1000):

```
    if a%2==1:
            print b
```

Output :

101,103,105,107,...........,997,999

24) Write a program to display multiplication table for a given number.

Answer : **Multiplication table :**

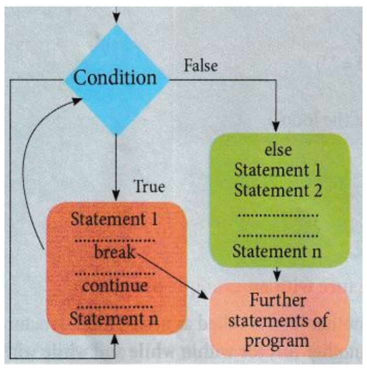
```
num =int(input("Enter the number: "))
print("multiplication Table of", num)
for i in range(1,11):
        print (num,"x",i, "= ",num*i )
```

Output :

```
Enter the number : 6
Multiplication Table of 6
6 x 1 = 6
6 x 2 = 12
6 x 3 = 18
6 x 4 = 24
6 x 5 = 30
6 x 6 = 36
6 x 7 = 42
6 x 8 = 48
6 x 9 = 54
6 x 10 = 60.
```

25) What are jump Statements in Python? Explain briefly about jump statements with its syntax and flowchart.

Answer : The jump statement in Python is used to unconditionally transfer the control from one part of the program to another. There are three keywords to achieve the jump statement in Python: break, continue, pass. The following flowchart illustrates the use of break and continue.



Break:

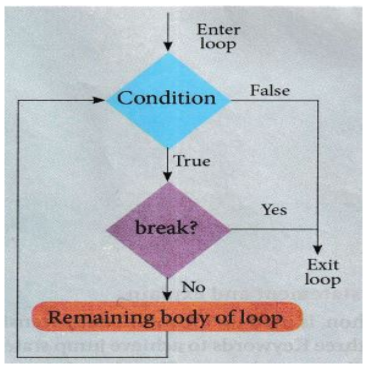
The break statement terminates the loop containing it. Control of the program flows to the statement immediately after the body of the loop.

A while or for loop will iterate till the condition is tested false, but one can even transfer the control out of the loop with help of a break statement. When the break statement is executed, the control flow of the program comes out of the loop and starts executing the segment of code after the loop Structure. If the break statement is inside a nested loop, the break will terminate the innermost loop.

Syntax:

break

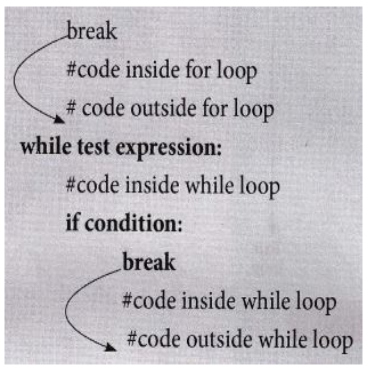
Flowchart



The working of break statement in for loop and while loop is shown in below:

for var in sequence:

if condition



```
    break
    #code inside for loop
    # code outside for loop
while test expression:
    #code inside while loop
    if condition:
        break
        #code inside while loop
        #code outside while loop
```

```
# Program to illustrate the use of break statement inside for loop.
for a word in "Jump Statement":
if word == 'e'
break
print (word, end = '')
else:
print(''End of the loop")
print ("\n End of the program")
```

Output:

```
Jump Stat
End of the program
```

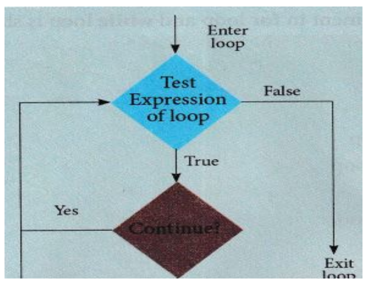
Continue

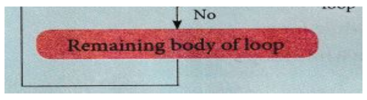
Continue statement unlike the break statement is used to skip the remaining part of a loop and start with next iteration.

Syntax:

continue

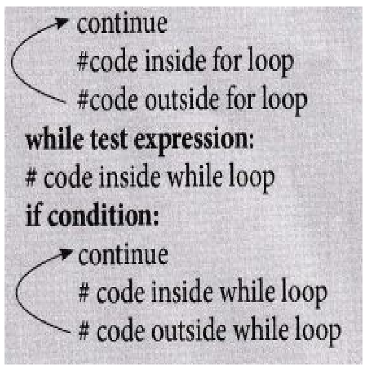
Flowchart:





The working continues statement in for and while loop is shown beloe.

```
for var in sequence:
#code inside for loop
if condition:
```



```
continue
#code inside for loop
#code outside for loop
while test expression:
# code inside while loop
if condition:
continue
# code inside while loop
# code outside while loop
```

# Program to illustrate the use of continue statement inside for loop.

```
for word in "Jump statement":
if word == "e":
continue
print (word, end =")
print("\n End of the program")
```

Output:

```
Jump statement
End of the program
```

Pass

Pass statement in Python programming is a null statement. Pass statement when executed by the interpreter it is completely ignored. Nothing happens when pass is executed, it results in no operation.

Pass statement can be used in 'if' clause as well as within loop construct when we do not want any statements or commands within that block to be executed.

Syntax:

```
pass
```

Program to illustrate the use of pass statement in for loop.

```
for val in "Computer":
pass
print ("End of the loop, loop structure will be built in future")
```

Output:

```
End of the loop, a loop structure will be built in the future.
```