Unit-2

Decision control Statement.

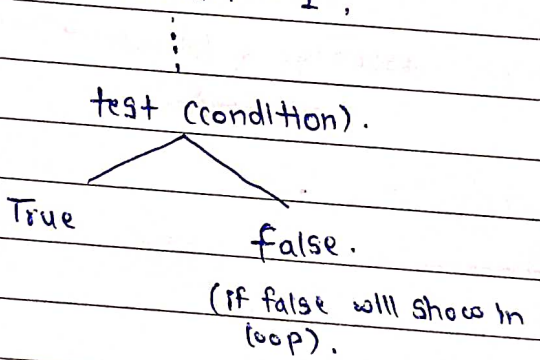
- Decision control statement determines the combined flow of set of instructions.

– there are 3 fundamentals of decision control statements

i] Sequence Control Statement:-

Statement 1.
Statement 2.
:
Statement n.

ii] Selection control statement:-



– In Selection control statement are followed by 4 statem

a) If
b) If else
c) Nested if
d) if elif else.

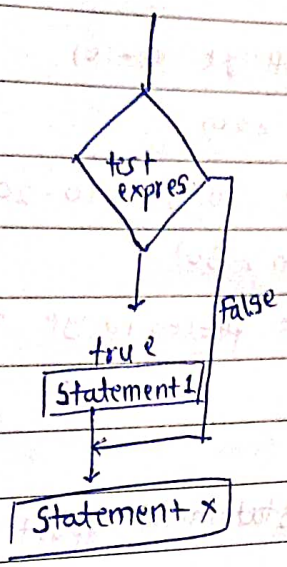
a] If:-

```
if test - expression):
    Statement 1
    :
    Statement n
Statement x.
```

if condition is true (statement)

if condition is false it will print x. (statement)

- If statement flow chart:-




```
Age = Int (input ("Enter your age :))
if Age >= 18;
    print ("You are eligible").
else;
    years = 18 - Age.
    Print (" Yes remaining to vote + statement (years + years) :
```

Q. To print 1st 10 even numbers using if else statement.

→
```
x = Int (input (" first 10 even numbers)).
if x/2 = 0; and x ≤ 20.
    Print (" 2, 4, 6, 8, 10, 12, 14, 16, 18, 20).
else;
    Print (" Not even number).
```

- Num = int (input ("Enter any no. from 0-30")).
if (num >= 0 and num < 10).
    print (" It is in range 0-10).
If num >= 10 and num < 20).
    print (" It is in the range 10-20").
If (num >= 20 and num < 30)
    print (" It is in the ~~proces~~ range 20-30").

- If elif else
It will test additional statement apart from initial statement.

- If (test expression);
    Statement block.
elif (test expression):
    Statement block:
else:
    Statement block.
Num = int (input ("Enter any no:"))
if n == 0;

1] x in range (10). (end)
print (x, end = " ").
→ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 (end is excluside dont count)

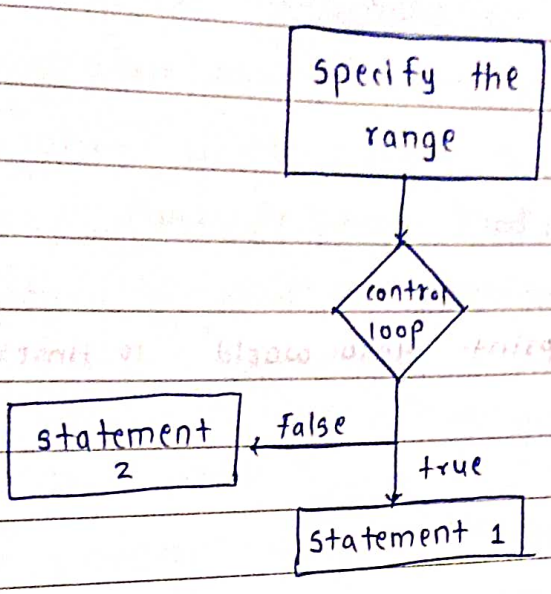
2] x in range (3, 10). (start, end)
→ 3, 4, 5, 6, 7, 8, 9.

3] x in range (3, 21, 2). (start, end, step)
→ 3, 5, 7, 9, 11, 13, 15, 17, 19.

for loop control in sequence, Range ()

Statement block - 1

range (Beg, End, steps)



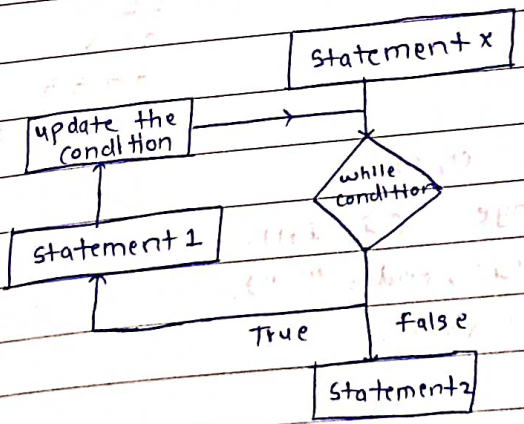
* While loop :-

Statement x

while (condition)

Statement 1.

statement 2.



Eg :-

```
i = 0
while (i <= 10).
print ('i', end " ")
i = 1 + i.
```

then output will be :-

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10.

Q. Write a program to print '*' form 8. 10 times.

→
```
i=0.
while (i<10).
print ('*', end = "*").
  i = 1 + i.
```

Q. Write a program to print 'Hello world' 10 times.

→
```
i=0.
while (i<10).
print 'Helloworld', "end" "):
  i = i+1.
```

* Nested for loop:-

Q. Write a program to print

```
1
22
333
4444
55555.
```

→
```
for i in range (1, 6)
  for 'j' in range (i, i+1).
    print ('i', end = " ").
    print (" \n").
```

* Nested while :-

```
while (i<5)
    print ("hi"), end = ", ")
        i = i+1
    while j <4
        Print ("hello" ) end =", " )
            j= j+1.
```

output.

hi
hello
hello
hello
hello
hi
hi
hi
hi.

* Break Statement :-

- Break statement is used to terminate the ~~near~~ execution of the nearegt enclosing loop in which appears.

- syntax for Break statement :-

```
break.
```

- It is used to exit a loop at any point.

```
i=0
j=0
while i<5
  Print (Hi)
  j = i+1
  while j<4.
  Print ("Hello").
    j = j+1
    break.
```

* Continous statement :-

```
j=0
j=0
while i<5.
print (Hi)
  i = i+1
while j<4.
  Print ("Hello").
continous j = j+1.
```

* Pass Statement:-

- As the name suggests pass statement simply does Nothing. The pass statement in python is used when a statement is required syntactically but you do not want any code or a comment to execute.
- It acts like null operation as nothing will happen as it is executed. Pass statement can also be used for writing empty loops.
- Pass is also used for empty control statement, functions & classes.

- Eg:-

```
i = 0
while (i <4):
    print ("less")
    i = i+1
else:
    print ("greater").
```

→ Output:

less
less
less
less
Greater.

- Lists = [20, 10, 20, 30, 2, 9.2, "hello"].
  [0] [1] [2] [3] [4] [5] [6]
  [-7] [-6] [-5] [-4] [-3] [-2] [-1]

Dictionary :-

python dictionary is an unordered collection of items while others compounds data types low only value as an element, a dictionary has a key value pair. Dictionary are opthmsed to retire the values when it is known. one or more key. value pairs.

Dictionary methods :-

keys () :- Removes a list cointaining the dictionary's keys.

Pop () :- Removes an element with a specified key.

popiten :- Removes the last insted key - value pair.