Basics of Programming

L03: Conditional Statements and Loops

Apr 09, 2010

Dr. Ram P Rustagi
Professor, CSE Dept
KRP, KSGI
rprustagi@ksit.edu.in

Resources and Acknowledgements

- Intro to Programming with C++
  - Abhiram Ranade, Prof CSE, IIT Bombay
- A first course in programming
  - https://introcs.cs.princeton.edu/python/home/
  - https://introcs.cs.princeton.edu/java/home/
- Python for everybody
  - https://www.py4e.com
- Web Applications for everybody
  - https://www.wa4e.com
- Turtle Graphics
  - https://docs.python.org/3/library/turtle.html

Review: Last Lecture

- Writing a program
  - Using term previously computed
    - e.g. e^1, e^x, $2/\pi$, `D(r)`
    - La Russe Algorithm for multiplication
- Program constructs
  - Basic Loop
  - Basics Functions

Concentric Circles

- Draw `5` concentric circles with a radius of `25px`
- Use the circle API
  - `circle(r)` # draws full circle
  - `circle(r, extent)`
    - e.g. `circle(r,180)` # draws semicircle

```
for i in range(n):
 penup();
 setpos(0,-25*(i+1));pendown()
 circle((i+1)*25)
```

Multiple Turtles

Q? What does following program draw?

```
t1=Turtle()
t2=Turtle()
t3=Turtle()
for i in range(6)
  t1.forward(100);
  t2.forward(100);
  t3.forward(100)
  t1.left(360/n)
  t2.left(360/n)
  t3.left(360/n)
```

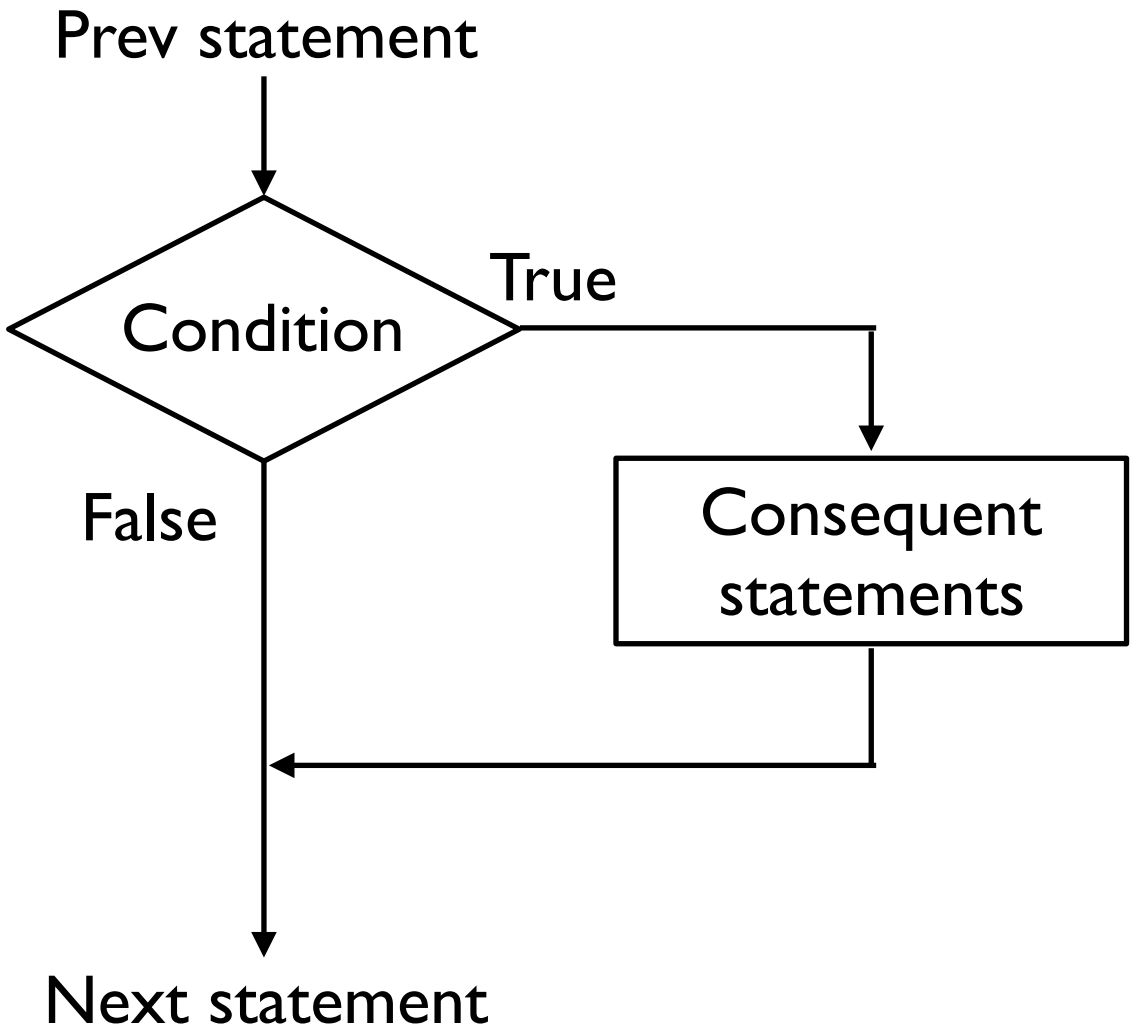
Assign Grades

- Write a program to assign grades using marks
  - If `marks ≥90`, grade '`A`'
  - if `80≤marks<90`, grade '`B`'
  - if `70≤marks<80`, grade '`C`'
  - if `60≤marks<70`, grade '`D`'
  - if `50≤marks<60`, grade '`E`'
  - if `marks<50`, grade '`F`'

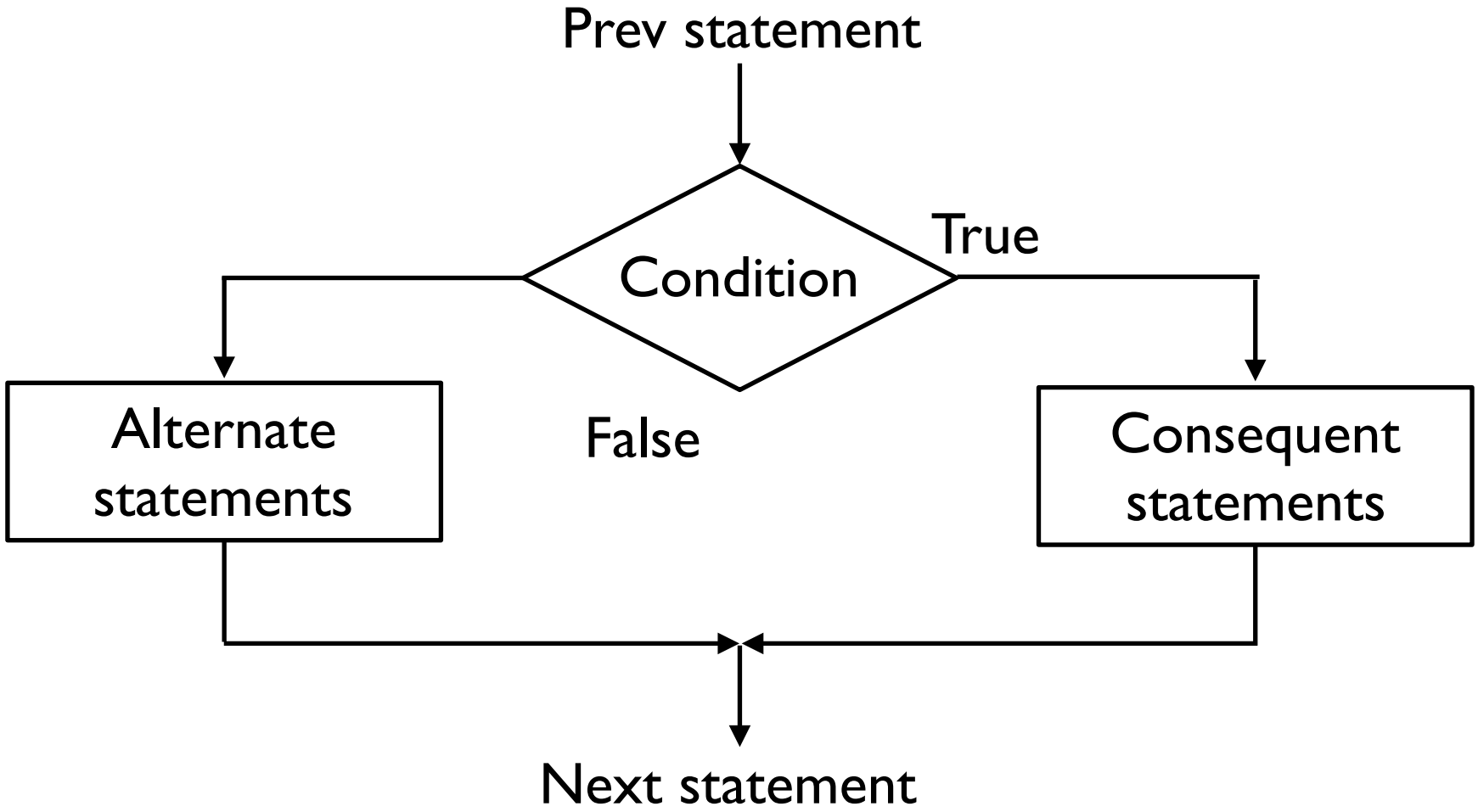
Conditional Statement: `if`

- Basic `if` statement
  - Solve a simple condition "yes"
- `if-else` statement
  - Better program to solve "yes"/"no"
- Most general `if` statement (`if-elif-else`)
  - To experess complex conditions
    - e.g. computing grades assignment
- Nested if statements
- `switch` statement
  - Another (better) way to express complex conditions

If Statement



If-else Statement



Most General If statement

- Block: a group of statement executed together
  - Languages define it in their way.
    - pitfalls in C/Java; pitfalls in python?
  - Together with `if` or `elif` or `else` condition
- Grades program

```
if marks >= 90:
   print(”A”)
elif marks >=80:
   print(”B”)
elif marks >=70:
   print(”C”)
   :
else:
   print(”F”)
```

Q: what happens if we replace **elif** by **if**

Complex Conditions

- Examples
  - `condition1 and condition2:`
  - `condition1 or condition2:`
  - `not condition`
- Consider program segment

```
for i in range(n):
 for j in range(n):
  if (i==0) or (j==0) or (i==n-1) or
  (j==n-1) or (i==j) or (i==n-1-j):
    print("*", end="")
  else:
    print(" ",end="")
  print("")
# end of program
```

Conditional Statement: switch

- Consider the case where input is an alphabet
  - For each alphabet value, you need to take different action.
  - A series of `if-elif-...elif-else` is required
    - Writing program becomes cumbersome:
      - Coding errors and debugging becomes complex.
    - A simple solution is to use `switch` statement (C/Java)
      - No `switch` statement support in python
      - To implement it in java, use dictionary (hash array)
      - definition

```
switcher = {'A': f1, 'B': f2, …}
```

      - invocation

```
fn=switcher.get(key,default_fn)…}
fn()
```

Loops

- Different languages support different looping variants
  - for loop
  - while loop
  - do while
  - repeat until
- Python support for iteration
  - for `i` in range(`n`):
  - for `i` in range(n_1,n_2)
  - for `i` in range(n_1,n_2,`step`)
  - while condition:
    - while `True`

Pre-termination of Loop

- Breaking the loop
  - break
- Continue to next iteration
  - continue
- Syntactic fulfillment requirement
  - pass
  - Example

```
for i in range(n):
    if i==1:
        pass # no need for computation
    else:
        # check for divisibility by i
```

Exercises

- A: Take following 3 inputs
  - Year : e.g. `2019, 2020`, etc.
  - Month name (e.g. Jan, Feb, …, Dec),
  - Date of the month (e.g. `1, 2, ..., 31`)

compute day of the year. Discard invalid inputs and consider leap year into the account

- For example:
  - `Feb 02, 2019` —> `33`
  - `Mar 03, 2020` —>`63` # leap year
  - `Apr 31, 2019` —> invalid input

Exercises

- B: Take n as input numbers and identify if it is prime number. Do not use boolean (True/False or its equivalent) variables
  - Use nested ifs

- C: Take 2 numbers and compute their GCD (Greatest Common Divisor)

- D: Take 4 numbers and sort them using 5 comparisons
  - Use nested ifs

Exercises

- E: Take `n` as input natural number and return the smallest palindrome larger than `n`

- F: Write a program that reads a sequence of integers (including negative numbers) e.g. as command line arguments

  $x_1, x_2, x_3, \ldots, x_n$

  – From this sequence, find the subsequence with maximum sum i.e. find $x_i, x_{i+1}, \ldots, x_j$ such that sum $x_i, x_{i+1}, \ldots, x_j$ is maximum.

  – Example: `2,-3, 1.5, -1, 3, -2, -3, 3`

    - The max sum is `3.5` (`1.5, -1, 3`)

Summary

- conditions
- no switch statement
- loops
- Exercises

Questions