

NPTEL MOOC

**PROGRAMMING,
DATA STRUCTURES AND
ALGORITHMS IN PYTHON**

Week 3, Lecture 3

Madhavan Mukund, Chennai Mathematical Institute

<http://www.cmi.ac.in/~madhavan>

Loops revisited

- * `for i in l:`

 - `. . .`

 - * Repeat body for each item in list `l`

- * `while condition:`

 - `. . .`

 - * Repeat body till `condition` becomes `False`

- * Sometimes we may want to cut short the loop

Search for value in a list

```
def findpos(l,v):  
    # Return first position of v in l  
    # Return -1 if v not in l  
  
    (found,i) = (False,0)  
    while i < len(l):  
        if l[i] == v:  
            (found,pos) = (True,i)  
  
    if not found:  
        pos = -1  
    return(pos)
```


Search for value in a list

```
def findpos(l,v):  
    # Return first position of v in l  
    # Return -1 if v not in l  
  
    (found,i) = (False,0)  
  
    while i < len(l):  
        if not found and l[i] == v:  
            (found,pos) = (True,i)  
  
    if not found:  
        pos = -1  
  
    return(pos)
```


Search for value in a list ...

- * A more natural strategy
 - * Scan list for value
 - * Stop scan as soon as we find the value
 - * If the scan completes without success, report -1

Search for value in a list ...

- * A more natural strategy

```
def findpos(l,v):
```

```
    for x in l:
```

```
        if x == v:
```

```
            # Exit and report position of x
```

```
    # Loop over, report -1 if we did not see x
```


Search for value in a list ...

- * A more natural strategy

```
def findpos(l,v)
```

```
    (pos,i) = (-1,0)
```

```
    for x in l:
```

```
        if x == v: # Exit, report position of x
```

```
            pos = i
```

```
            break
```

```
            i = i+1
```

```
# If pos not reset in loop, pos is -1
```

```
return(pos)
```


Search for value in a list ...

- * A more natural strategy

```
def findpos(l,v)
    pos = -1
    for i in range(len(l)):
        if l[i] == v: # Exit, report position
            pos = i
            break

    # If pos not reset in loop, pos is -1
    return(pos)
```


Search for value in a list ...

- * A loop can also have an `else`: — signals normal termination

```
def findpos(l,v)
    for i in range(len(l)):
        if l[i] == v: # Exit, report position
            pos = i
            break
    else:
        pos = -1 # No break, v not in l
    return(pos)
```


Summary

- * Can exit prematurely from loop using `break`
- * Applies to both `for` and `while`
- * Loop also has an `else:` clause
 - * Special action for normal termination