

Εισαγωγή στην Επιστήμη των Υπολογιστών

Λίστα (1)

```
>>> numbers = [17, 123]
```

```
>>> numbers[1] = 5
```

```
>>> print numbers
```

```
[17, 5]
```

Λίστα (2)

```
>>> cheeses = ['Cheddar', 'Edam', 'Gouda']
```

```
>>> 'Edam' in cheeses
```

```
True
```

```
>>> 'Brie' in cheeses
```

```
False
```

Λίστα (3)

```
>>> a = [1, 2, 3]
```

```
>>> b = [4, 5, 6]
```

```
>>> c = a + b
```

```
>>> print c
```

```
[1, 2, 3, 4, 5, 6]
```

Λίστα (4)

```
>>> t = ['a', 'b', 'c', 'd', 'e', 'f']
```

```
>>> t[1:3]
```

```
['b', 'c']
```

```
>>> t[:4]
```

```
['a', 'b', 'c', 'd']
```

```
>>> t[3:]
```

```
['d', 'e', 'f']
```

```
>>> t[:]
```

```
['a', 'b', 'c', 'd', 'e', 'f']
```

Λίστα (5)

```
>>> t = ['a', 'b', 'c']
```

```
>>> t.append('d')
```

```
>>> print t
```

```
['a', 'b', 'c', 'd']
```

```
>>> t1 = ['a', 'b', 'c']
```

```
>>> t2 = ['d', 'e']
```

```
>>> t1.extend(t2)
```

```
>>> print t1
```

```
['a', 'b', 'c', 'd', 'e']
```

Λίστα (6)

```
names = [ "Dave", "Mark", "Ann", "Phil" ]  
a = names[2]    # "Ann"  
names[0] = "Jeff " # Το πρώτο item γίνεται "Jeff "  
names.append("Paula") # Προσθήκη στο τέλος  
names.insert(2, "Thomas") # Προσθήκη ως 3ο  
b = names[0:2] # Returns [ "Jeff", "Mark" ]  
c = names[2:]  
# Returns [ "Thomas", "Ann", "Phil", "Paula" ]
```

Λίστα (7)

```
>>> t = ['d', 'c', 'e', 'b', 'a']
```

```
>>> t.sort()
```

```
>>> print t
```

```
['a', 'b', 'c', 'd', 'e']
```

```
>>> t = [1, 2, 3]
```

```
>>> sum(t)
```

```
6
```


Λίστα (8)

b = [1,2,3] + [4,5] # Αποτέλεσμα [1,2,3,4,5]

a = [1,"Dave",3.14, ["Mark", 7, 9, [100,101]], 10]

a[1] # "Dave"

a[3][2] # 9

a[3][3][1] # 101

Λίστα (9)

```
b = ["Dave", "Mark", "Ann", "Phil"]  
for name in b:  
    print name
```

Dave

Mark

Ann

Phil

Χρήση while

```
fruit = ['Banana', 'Apple', 'Orange']
```

```
index = 0
```

```
while index < len(fruit):
```

```
    letter = fruit[index]
```

```
    print letter
```

```
    index = index + 1
```

Χρήση συνάρτησης

```
def capitalize_all(t):  
    res = []  
    for s in t:  
        res.append(s.capitalize())  
    return res
```

Άθροισμα

```
total = 0
```

```
for i in range(0,len(items),1):
```

```
    total += items[i]
```

Εξαγωγή στοιχείου από τη λίστα

```
>>> t = ['a', 'b', 'c']
```

```
>>> x = t.pop(1)
```

```
>>> print t
```

```
['a', 'c']
```

```
>>> print x
```

```
b
```

Διαγραφή στοιχείου από τη λίστα

```
>>> t = ['a', 'b', 'c']
```

```
>>> del t[1]
```

```
>>> print t
```

```
['a', 'c']
```

```
>>> t = ['a', 'b', 'c']
```

```
>>> t.remove('b')
```

```
>>> print t
```

```
['a', 'c']
```

Μετατροπή String σε λίστα

```
>>> s = 'spam'
```

```
>>> t = list(s)
```

```
>>> print t
```

```
['s', 'p', 'a', 'm']
```


Μετατροπή String σε λίστα (2)

```
>>> s = 'pining for the fjords'
```

```
>>> t = s.split()
```

```
>>> print t
```

```
['pining', 'for', 'the', 'fjords']
```

```
>>> s = 'spam-spam-spam'
```

```
>>> delimiter = '-'
```

```
>>> s.split(delimiter)
```

```
['spam', 'spam', 'spam']
```

Μετατροπή λίστας σε String

```
>>> t = ['pinning', 'for', 'the', 'fjords']
```

```
>>> delimiter = ' '
```

```
>>> delimiter.join(t)
```

```
'pinning for the fjords'
```

Δείκτες

```
>>> a = [1, 2, 3]
```

```
>>> b = a
```

```
>>> b is a
```

```
True
```

```
>>> b[0] = 17
```

```
>>> print a
```

```
[17, 2, 3]
```

Δείκτες (2)

```
>>> a = [3,4,5]
```

```
>>> b = [a]
```

```
>>> c = 4*b
```

```
>>> c
```

```
[[3, 4, 5], [3, 4, 5], [3, 4, 5], [3, 4, 5]]
```

```
>>> a[0] = -7
```

```
>>> c
```

```
[[ -7, 4, 5], [ -7, 4, 5], [ -7, 4, 5], [ -7, 4, 5]]
```

Χρήση λιστών

```
>>> t1 = [1, 2]
>>> t2 = t1.append(3)
>>> print t1
[1, 2, 3]
>>> print t2
None
>>> t3 = t1 + [3]
>>> print t3
[1, 2, 3]
```

Χρήση for (1)

```
fruits = ['banana', 'apple', 'mango']
```

```
for fruit in fruits:
```

```
    print 'This fruit is a', fruit
```

```
This fruit is a banana  
This fruit is a apple  
This fruit is a mango
```

Χρήση for (2)

```
a=['Mary','had','a','little','lamb']
```

```
for i in range(len(a)):
```

```
    print i, a[i]
```

```
0 Mary  
1 had  
2 a  
3 little  
4 lamb
```

Αναζήτηση

```
def find(target,items):  
    found = False  
    i = 0  
    while (not(found) and i < len(items)):  
        if (items[i] == target):  
            found = True  
        i += 1  
    return found
```


Αναζήτηση (II)

```
def find(target,items):  
    for i in range(0,len(items),1):  
        if (items[i] == target):  
            return True  
    return False
```

Εξαγωγή άρτιων

```
def extractEvens(items):  
    evens = []  
    for i in range(0, len(items), 1):  
        if (items[i] % 2 == 0):  
            evens = evens + [items[i]]  
    return evens
```

Εξαγωγή άρτιων (2)

```
>>> extractEvens([4,2,5,2,7,0,8,3,7])
```

```
[4, 2, 2, 0, 8]
```

```
>>> extractEvens([1,3,5,7,9])
```

```
[]
```

Εύρεση μεγαλύτερου

```
largest = items[0]
for i in range(1,len(items),1):
    if (items[i] > largest):
        largest = items[i]
```

Εύρεση της θέσης του μεγαλύτερου

```
ilargest = 0
```

```
for i in range(1,len(items),1):
```

```
    if (items[i] > items[ilargest]):
```

```
        ilargest = i
```

Ταξινομημένη ένωση λιστών

```
def merge(list1,list2):  
    list3 = []  
    i = 0  
    j = 0  
    while (i < len(list1) and j < len(list2)):  
        if (list1[i] < list2[j]):  
            list3 = list3 + [list1[i]]  
            i = i + 1  
        else:  
            list3 = list3 + [list2[j]]  
            j = j + 1  
    return list3 + list1[i:] + list2[j:]
```

fibonacci

```
def fib2(n):  
    result = []  
    a, b = 0, 1  
    while a < n:  
        result.append(a)  
        print a,  
        a, b = b, a+b  
    return result
```

Χρήση λίστας

`a = [1,2,3,4,5]`

`a[1] = 6 # a = [1,6,3,4,5]`

`a[2:4] = [10,11] # a = [1,6,10,11,5]`

`a[3:4] = [-1,-2,-3] # a = [1,6,10,-1,-2,-3,5]`

`a[2:] = [0] # a = [1,6,0]`

Χρήση λίστας (2)

```
values = [1, 100, 45, 23, 73, 37, 69 ]  
clamped = [x if x < 50 else 50 for x in values]  
print(clamped) # [1, 50, 45, 23, 50, 37, 50]
```

Χρήση λίστας (3)

```
a = [1, 2, 3, 4, 5]
```

```
def square(items):
```

```
    for i,x in enumerate(items):
```

```
        items[i] = x * x
```

```
square(a) # a γίνεται [1, 4, 9, 16, 25]
```

Χρήση λίστας (4)

```
a = [-3,5,2,-10,7,8]
```

```
b = 'abc'
```

```
c = [2*s for s in a]
```

```
# c = [-6,10,4,-20,14,16]
```

```
d = [s for s in a if s >= 0]
```

```
# d = [5,2,7,8]
```

```
e = [(x,y) for x in a  
      for y in b  
      if x > 0 ]
```

```
# e = [(5, 'a'), (5, 'b'), (5, 'c'),  
      (2, 'a'), (2, 'b'), (2, 'c'),  
      (7, 'a'), (7, 'b'), (7, 'c'),  
      (8, 'a'), (8, 'b'), (8, 'c')]
```

Μαθηματικές συναρτήσεις

```
>>> from decimal import *
```

```
>>> c = [Decimal("4.5"), Decimal("3"),  
Decimal("1.23e3")]
```

```
>>> sum(c)
```

```
Decimal("1237.5")
```

```
>>> [10*x for x in c]
```

```
[Decimal("45.0"), Decimal("30"),  
Decimal("1.230e4")]
```