

ΠΑΡΑΔΕΙΓΜΑ ΜΕ ΔΥΑΔΙΚΗ ΑΝΑΖΗΤΗΣΗ

1) Να γράψετε πρόγραμμα σε Python το οποίο:

α) να διαβάζει τους αριθμούς μητρώου και τους βαθμούς 20 μαθητών στο μάθημα της ιστορίας και να τους καταχωρίζει σε δύο λίστες AM και BA αντίστοιχα

β) να υπολογίζει και να εμφανίζει το ΜΟ όλων των βαθμών των μαθητών

γ) να υπολογίζει και να εμφανίζει πόσοι μαθητές είχαν βαθμό κάτω από το ΜΟ της τάξης

δ) Θεωρώντας ότι μόνο ένας μαθητής έχει το βαθμό 14 με τη βοήθεια της binarysearch που επιστρέφει τη θέση να βρείτε τον AM του μαθητή με αυτό το βαθμό.

Σημείωση: Θεωρούμε ότι οι λίστες είναι ταξινομημένες κατά αύξουσα σειρά ως προς το βαθμό.

ΠΑΡΑΔΕΙΓΜΑ 1

```
def binarysearch( BA, v ) :  
  
    first = 0  
  
    last = len(BA) - 1  
  
    F = False  
  
    while F == False and first <= last:  
  
        mid = ( first + last ) // 2  
  
        if BA[ mid ] == v :  
  
            F = True  
  
            if v > BA[ mid ] :  
  
                first = mid + 1  
  
            if v < BA[ mid ] :  
  
                last = mid-1  
  
    return mid
```

2^{ος} Τρόπος

```
if BA[ mid ] == v :  
    found = True  
  
elif v > BA[ mid ] :  
    first = mid + 1  
  
else:  
    last = mid-1
```

```
AM=[]  
BA=[]  
for x in range(20):  
    am=str(raw_input('Dose AM'))  
    v=input("Dose Vathmo")  
    AM.append(am)  
    BA.append(v)
```

Εισαγωγή Δεδομένων
στις λίστες

```
S = 0.0
```

```
for x in BA:
```

```
    S=S+x
```

```
MO=S/20
```

```
print "Mosos Oros Misthon = ", MO
```

Υπολογισμός
Μέσου Όρου

```
c=0
```

```
for x in BA:
```

```
    if x < MO:
```

```
        c = c+1
```

```
print c
```

Μετράει πόσοι βαθμοί
είναι μικρότεροι από τον
Μέσο Όρο

```
thesi = binarysearch( BA, 14 )
```

```
print AM[ thesi ]
```

Καλεί την Συνάρτηση της
Διαδικής Αναζήτησης

ΑΣΚΗΣΗ

Να γράψετε πρόγραμμα σε Python το οποίο:

α) να διαβάζει το επώνυμο και το όνομα των μαθητών μιας τάξης και το σταθερό τηλέφωνο του καθενός. Τα στοιχεία αυτά καταχωρούνται στις λίστες **EP,ON** και **THL** αντίστοιχα. Η διαδικασία αυτή να τερματίζεται όταν δοθεί ως Επώνυμο η λέξη «**TELOS**».

β) στη συνέχεια να γράψετε τη **συνάρτηση** `def binarySearch(lista, key)` η οποία να επιστρέφει τη θέση του `key` μέσα στο `lista`

γ) να διαβάζει ένα επώνυμο και με τη βοήθεια της παραπάνω (β) συνάρτησης να εμφανίζει το σταθερό τηλέφωνο του μαθητή με αυτό το επώνυμο ή το μήνυμα “δεν υπάρχει” αν δεν υπάρχει αυτό το επώνυμο.

Υποδείξεις: Θεωρούμε ότι όλοι οι μαθητές έχουν διαφορετικά επώνυμα.

ΣΗΜΕΙΩΣΗ: οι τρεις λίστες είναι ταξινομημένες σε **αύξουσα** σειρά ως προς το επώνυμο.

ΛΥΣΗ

```
def binarysearch( BA, v ) :  
    first = 0  
    last = len(BA)-1  
  
    F = False  
    while F == False and first <= last:  
  
        mid = ( first + last ) // 2  
  
        if BA[ mid ] == v :  
            F = True  
        if v > BA[ mid ] :  
            first = mid + 1  
        if v < BA[ mid ] :  
            last = mid-1  
  
    if F==True:  
        return mid  
    if F==False:  
        return -1
```

```
EP=[]
ON=[]
THL=[]

ep=str(raw_input("Dose Eponimo"))
while ep!="TELOS":

    on=str(raw_input("Dose onoma"))
    thl=str(raw_input("Dose tilefono"))

    EP.append(ep)
    ON.append(on)
    THL.append(thl)

    ep=str(raw_input("Dose NEO Eponimo"))

ep=raw_input("Dose epinimo ?")

thesi=binarysearch( EP, ep )

if thesi !=-1:
    print THL[thesi]

else:
    print "DEN YPARXEI"
```