



A1.

- α) ΣΩΣΤΟ
- β) ΛΑΘΟΣ
- γ) ΛΑΘΟΣ
- δ) ΛΑΘΟΣ
- ε) ΣΩΣΤΟ

A2.

- 1) β
- 2) στ
- 3) δ
- 4) γ
- 5) α

B1.

```
def trim_a(s1):  
    s2 = ''  
    for x in s1:  
        if x != 'a' and x != 'A':  
            s2 = s2 + char  
    return s2
```

B2.

- α) 73, 181, 145, 98
- β) 73, 29, 12

B3.

```
i=0  
while i<10:  
    j = 10  
    while j > -1:  
        print i*j  
        j -= 1  
    i += 1
```

Γ.

```
max = -1
pl = 0
name = raw_input()
while name != 'ΤΕΛΟΣ':
    pl += 1
    s=0
    pl1 = 0
    for i in range(10):
        vath = int(input())
        while vath < 1 or vath > 20:
            vath = int(input())
        s=s+vath
        if vath>=12:
            pl1 += 1
    mo= s / 10.0
    print (mo)
    if mo > 15 and pl1 == 10:
        print ('ΠΡΟΚΡΙΝΕΤΑΙ ΣΤΗΝ ΕΠΟΜΕΝΗ ΦΑΣΗ')
        pl2 = pl2 + 1
    if mo > max:
        max = mo
    name = raw_input()
    pos = float(pl2) / pl * 100
print (max, pos)
```

Δ.

```
ON = []
f1 = open('branch.txt','r')
for line in f1:
    ON.append(line)
f1.close()
S_POSO = []
for i in range(len(ON)):
    s = 0
    for j in range(30):
        inc = float(input())
        s = s + inc
    S_POSO.append(s)
s1 = 0
for x in S_POSO:
    s1 = s1 + x
mo = s1 / len(S_POSO)
print (mo)
p = 0
for x in S_POSO:
    if x >= mo:
        p += 1
print(p)

N = len(S_POSO)
for i in range(N-1):
    for j in range(N-1,i,-1):
        if (S_POSO[j] > S_POSO[j-1]) or (S_POSO[j] == S_POSO[j-1] and ON[j] < ON[j-1]):
            S_POSO[j], S_POSO[j-1] = S_POSO[j-1], S_POSO[j]
            ON[j], ON[j-1] = ON[j-1], ON[j]
```