

ΦΥΣ 140 – LAB06 - 15 Λεπτά

(α) [3μ] Το περιεχόμενο του αρχείου names.txt είναι το ακόλουθο:

```
Afrofiti  
Marina  
Sotiroulla
```

Ποιο από τα παρακάτω τμήματα κώδικα θα τυπώσει όλα τα ονόματα του names.txt;

(I)

```
names = open("names.txt", "r")  
for line in names:  
    print(names)
```

Δεν θα τυπώσει τα ονόματα
γιατί το object names είναι
τύπου file_handler

(II)

```
names = open("names.txt", "r")  
for line in names:  
    print(line)
```

Θα τυπώσει τα ονόματα

(III)

```
names = open("names.txt", "r")  
for line in names:  
    print("line")
```

Δεν θα τυπώσει τα ονόματα
γιατί θα τυπώσει το string
"line"

(β) [2μ] Ποια λέξη κλειδί θα χρησιμοποιήσετε για να προσδιορίσετε το κώδικα που θα πρέπει να εκτελεστεί σε περίπτωση που υπάρχει error όταν εκτελεστεί το τμήμα της δομής **try**;

(A) catch

(B) exception

(Γ) except

(Δ) error

Η δομή είναι της μορφής:
try:
... (block εντολών)
except:
... (block εντολών)

(γ) [2μ] Πόσα λάθη υπάρχουν στο ακόλουθο τμήμα κώδικα; Το πρόγραμμα πρέπει να ανοίγει ένα file σε read-only mode, να διαβάζει και να τυπώνει κάθε γραμμή και τέλος να κλείνει το αρχείο.

```
def print_content(file):  
    file_obj = open(file)  
    for line in "file_obj" :  
        print(line_obj)
```

(A) 1.

(B) 2

(Γ) 3

(Δ) 4

(1) Έπρεπε να είναι file_obj και όχι "file_obj"
(2) Έπρεπε να είναι print(line) και όχι print(file_obj)
(3) Δεν υπάρχει file_obj.close()

(δ) [3μ] Τι θα τυπώσει το ακόλουθο τμήμα κώδικα:

```
Test = [2*x for x in range(101) if x%2==1 if x%5==0]  
print(Test)
```

Η συνθήκη στο list comprehension επιλέγει τους αριθμούς από 0 έως 100 που είναι περιττοί και πολλαπλάσια του 5. Κάθε επιλεγμένος αριθμός πολλαπλασιάζεται με 2 πριν εισαχθεί στη list Test. Η λίστα θα είναι [10,30,50,70,90,110,130,150,170,190]

- (ε) [5μ] Γράψτε το τμήμα κώδικα που γεμίζει μια λίστα με τιμές στο διάστημα [0,20] με βήμα 0.5 (δηλαδή τα στοιχεία της λίστας θα είναι [0, 0.5, 1, 1.5, 2,...,19.5, 20] χρησιμοποιώντας μόνο:
- (i) [1μ] μια δομή for loop
 - (ii) [2μ] list comprehension
 - (iii)[2μ] μια συνάρτηση και τη μέθοδο map

```
#!/usr/bin/python3

xup = 20; xlo = 0; dx = 0.5
nsteps = int( (xup - xlo)/dx ) # Αριθμός βημάτων για να πάμε από το 0 -> 20
# (a) for loop
a=[]
for i in range(nsteps+1) :      # Τα ορίσματα της μεθόδου range είναι μόνο τύπου int
    a += [ i*dx ]
print(a)
# (b) list comprehension
a = [ x*dx for x in range(nsteps+1) ]
print(a)
# (c) function and map
def oper(x):
    return x*dx
a=list(map(oper,range(nsteps+1)))
print(a)
quit()
```