

Εισαγωγή στη γλώσσα προγραμματισμού



Ενότητα 4: Βιβλιοθήκες και εξαιρέσεις

Μάθημα 15.

Διαχείριση εξαιρέσεων try/except

15 try/except

Exception : η κατάσταση σφάλματος στην οποία εισέρχεται ένα πρόγραμμα.

```
>>> a = 10
>>> b = 0
>>> print(a/b)
Traceback (most recent call last):
  File "<pyshell#8>", line 1, in <module>
    print(a/b)
ZeroDivisionError: division by zero
```

Η δομή **try/except** μας επιτρέπει να αντιμετωπίσουμε αυτά τα σφάλματα ώστε να μην διακοπεί το προγράμμα μας

*Exception τύπου
ZeroDivisionError*

15. Δομή try / except / finally



try:

εντολές # εδώ ελέγχεται το σφάλμα

...

except <τύπος σφάλματος -1> :

εντολές

except <τύπος σφάλματος -2> :

εντολές

finally :

εντολές

try:

print(a/b)

except ZeroDivisionError :

print('όχι διαίρεση με μηδέν!')

finally :

print("a=", a, "b=", b)

15. Εκτύπωση σφάλματος



try:

εντολές # εδώ ελέγχεται το σφάλμα

...

except <τύπος σφάλματος -1> **as** e :
 εντολές

```
>>> try:
        print(10/0)
except ZeroDivisionError as e:
    print(e)
```

```
division by zero
```

```
>>>
```

15. Δομή try / except / finally



Απαραίτητη η try/except σε διεπαφές του προγράμματος με τρίτους (πχ):

- input από τον χρήστη
- άνοιγμα αρχείου
- πρόσβαση στο διαδίκτυο

Άσκηση



Πώς θα χρησιμοποιήσουμε τη δομή try/except για να ελέγξουμε αν ο χρήστης μας έδωσε αριθμό;

```
num = " "  
while type(num) is str :  
    num = input('δώσε αριθμό:')  
    try:  
        num = float(num)  
    except ValueError :  
        print('Παρακαλώ δώστε αριθμό')  
# συνέχεια προγράμματος ...  
print ('ευχαριστώ')
```

Άσκηση



Να ξαναγράψετε την άσκηση 9.3 που ζητούσε 2 ακεραίους και υπολόγιζε το πηλίκο και το υπόλοιπο της διαίρεσής τους έτσι ώστε να αποφεύγονται όλες οι περιπτώσεις εσφαλμένων τιμών του χρήστη. Χρησιμοποιήστε τη δομή try/except