

Εργαστήριο 1

Εισαγωγή στον Προγραμματισμό Υπολογιστών

Περιεχόμενα

- Εκφράσεις
 - Ενθεματικοί τελεστές, εκφράσεις κλήσεις
 - Ειδικοί χαρακτήρες συμβολοσειρών
 - Λογικές εκφράσεις
 - Συγκρίσεις
 - Τελεστές ανά bit (bitwise)
 - Εντολή ανάθεσης
- Ανάπτυξη προγράμματος σε text editor

Εκφράσεις

- Υπολογίστε την τιμή $1 + \frac{2}{2 - \frac{3}{3 \cdot 4^{\frac{2}{3}}}}$ εισάγοντας στο διαδραστικό περιβάλλον της Python:

- έκφραση με ενθεματικούς τελεστές (χωρίς εκφράσεις κλήσης)

```
>>> _____
```

- έκφρασεις κλήσης χωρίς ενθεματικούς τελεστές

```
>>> from operator import add, truediv, sub, mul, pow
```

```
>>> _____
```

Εκφράσεις

```
sub(add(2, floordiv(5, add(1,1))), pow(mul(3, 2), 2))
```

- Γράψτε μια ισοδύναμη έκφραση χρησιμοποιώντας μόνο ενθεματικούς τελεστές (όχι εκφράσεις κλήσης)

Εκφράσεις

$$3 / (3 * 2)$$

$$3 / 3 * 2$$

- Γράψτε ισοδύναμες εκφράσεις χρησιμοποιώντας μόνο εκφράσεις κλήσης (χωρίς ενθεματικούς τελεστές)

Εκφράσεις

- Ενθεματικοί τελεστές συμβολοσειρών

```
>>> x, y = '\\', '/'
```

```
>>> z = _____
```

```
>>> print(z)
```

```
\\ / \\ / \\ /
```

```
Αναθέσεις  
>>> x, y = 1, 2  
>>> x  
1  
>>> y  
2
```

Χρησιμοποιήστε έκφραση χωρίς τους χαρακτήρες \, /

Ειδικοί χαρακτήρες συμβολοσειρών

```
>>> print('_____')
```

```
h      w
```

```
e      o
```

```
l      r
```

```
l      l
```

```
o      d
```

- Αλλαγή γραμμής (*newline*) `\n`

```
>>> print('abcdef\nghij')
abcdef
ghij
```

- Στοίχιση (*tab*) `\t`

```
>>> print('abcdef\tghij')
abcdef  ghij
```

- Επιστροφή κεφαλής (*carriage return*) `\r`

```
>>> print('abcdef\rghij')
ghijef
```

- Κύληση γραμμής (*line feed*) `\f`

```
>>> print('abcdef\fghij')
abcdef
      ghij
```

- Οπισθοδρόμηση (*backspace*) `\b`

```
>>> print('abcdef\bghij')
abcdeghij
```

- Ήχος ειδοποίησης (*bell*) `\a`

```
>>> print('abcdef\aghij')
abcdefghij
```

Ειδικοί χαρακτήρες συμβολοσειρών

```
>>> print ( _____ )
```

h

e

l

l w

o

r

l

d

- *Αλλαγή γραμμής (newline) \n*

```
>>> print ('abcdef\nghij')
```

abcdef
ghij
- *Κύληση γραμμής (line feed) \f*

```
>>> print ('abcdef\fghij')
```

abcdef
ghij
- *Στοίχιση (tab) \t*

```
>>> print ('abcdef\tghij')
```

abcdef ghij
- *Οπισθοδρόμηση (backspace) \b*

```
>>> print ('abcdef\bghij')
```

abcdeghij
- *Έχος ειδοποίησης (bell) \a*

```
>>> print ('abcdef\bghij')
```

abcdefghij
- *Επιστροφή κεφαλής (carriage return) \r*

```
>>> print ('abcdef\rghij')
```

ghijef

Λογικές εκφράσεις

- Γράψτε λογική έκφραση ώστε η `z` να έχει τιμή `True` μόνο εάν είτε `x` είναι `True` είτε `y` είναι `True` αλλά όχι ταυτόχρονα. (Αποκλειστικό ή – Exclusive OR (XOR), πχ.

```
>>> x = True
```

```
>>> y = False
```

```
>>> z = _____
```

```
>>> z
```

```
True
```

Συγκρίσεις

- Γράψτε λογική έκφραση ώστε η `z` να έχει τιμή `True` μόνο εάν η τιμή του `x` είναι άρτιος αριθμός ανάμεσα στο 6 και στο 88, πχ,

```
>>> x = 12
```

```
>>> z = _____
```

```
>>> z
```

```
True
```

Τελεστές ανά bit (bitwise)

- Συμπληρώστε τα κενά με τον σωστό bitwise τελεστή

>>> 1011 ___ 1

2022

>>> 2022 ___ 1

1011

τελεστής	πράξη ανά bit
&	AND
	OR
^	XOR
~	NOT
<<	Shift left by
>>	Shift right by

Τελεστές ανά bit (bitwise)

- Συμπληρώστε τα κενά με τον σωστό τελεστή

```
>>> 0b1010 & _____
```

```
2
```

```
>>> _____ | 0b101
```

```
7
```

```
>>> bin(7)
```

```
'0b111'
```

```
>>> bin(0b1010 ^ _____)
```

```
'0b1111'
```

τελεστής	πράξη ανά bit
&	AND
	OR
^	XOR
~	NOT
<<	Shift left by
>>	Shift right by

Εντολή ανάθεσης

```
>>> x = 13
```

```
>>> 2 * x
```

```
26
```

- Συμπληρώστε τα κενά που ακολουθούν με την *ίδια ακριβώς* εντολή

```
>>> x = 1
```

```
>>> x = _____
```

```
>>> x
```

```
2
```

```
>>> x = _____
```

```
>>> x
```

```
3
```

```
>>> x = _____
```

```
>>> x
```

```
4
```

Εντολή ανάθεσης

- Συμπληρώστε τα κενά που ακολουθούν με την *ίδια ακριβώς* εντολή

```
>>> x = '*'
```

```
>>> x = _____
```

```
>>> x
```

```
'*_*'
```

```
>>> x = _____
```

```
>>> x
```

```
'*_*_*_*'
```

```
>>> x = _____
```

```
>>> x
```

```
'*_*_*_*_*_*_*_*'
```

Ανάπτυξη προγράμματος σε text editor

1. Κατεβάστε το αρχείο `program.py` από τα Έγγραφα/Εργαστήρια/Εργαστήριο 1 του eclass στο Desktop (επιφάνεια εργασίας)
2. Εκκινήστε κάποιον text editor, πχ, Notepad++
 - Αν δεν υπάρχει στον Η/Υ, θα πρέπει να το εγκαταστήσετε από <https://notepad-plus-plus.org/download/v7.5.8.html>
3. Φορτώστε σε αυτόν (από μενού: File -> Open) το πρόγραμμα Python στο `c:\Users_____\Desktop\myfolder\program.py`
4. Αντικαταστήστε τα κενά `__` στο πρόγραμμα με ό,τι θέλετε και σώστε τις αλλαγές με File->Save (από μενού)
5. Από γραμμή εντολών MSDOS:

```
> python c:\users\_____\Desktop\program.py
```

 - Επιβεβαιώστε ότι εμφανίστηκαν τα αναμενόμενα αποτελέσματα