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

1. έκφραση με ενθεματικούς τελεστές (χωρίς εκφράσεις κλήσης)

```
>>> _____________________________
```

2. έκφρασεις κλήσης χωρίς ενθεματικούς τελεστές

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

Εκφράσεις

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

1. έκφραση με ενθεματικούς τελεστές (χωρίς εκφράσεις κλήσης)

```
>>> 1 + 2/(2-3/(3*4**(2/3)))
2.247544102606856
```

2. έκφρασεις κλήσης χωρίς ενθεματικούς τελεστές

```
>>> from operator import add, truediv, sub, mul, pow
>>> add(1, truediv(2, sub(2, truediv(3, mul(3, pow(4, truediv(2, 3)))))))
2.247544102606856
```

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

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

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

```
2 + 5 // (1 + 1) - (3 * 2) ** 2
```

Εκφράσεις

```
3 / (3 * 2)
3 / 3 * 2
```

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

Εκφράσεις

```
3 / (3 * 2)
3 / 3 * 2
```

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

```
truediv(3, mul(3, 2))
mul(truediv(3, 3), 2)
```

Εκφράσεις

- Ενθεματικοί τελεστές συμβολοσειρών

```
>>> x, y = '\\', '/'
>>> z = _____________________________
>>> print(z)
\/\/ \/\/ \/\/
```

Αναθέσεις

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

Χρησιμοποιήστε έκφραση *χωρίς* τους χαρακτήρες \, /

Εκφράσεις

- Ενθεματικοί τελεστές συμβολοσειρών

```
>>> x, y = '\\', '/'
>>> z = 3 * (2 * (x + y) + ' ')
>>> 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\tw\ne\to\nl\tr\nl\tl\no\td')
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
```
- *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\bghij')
abcdefghij
```

Ειδικοί χαρακτήρες συμβολοσειρών

```
>>> print('h\fe\fl\fl w\f\b\bo\f\b\br\f\b\bl\f\b\bd')
h
 e
  l
   l w
     o
   r
  l
 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
```

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
```

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 = (x and not y) or (not x and y)
>>> 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
```

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 = x % 2 == 0 and x >= 6 and x <= 88
>>> z
True
```

Τελεστές ανά bit (bitwise)

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

- Συμπληρώστε τα κενά με τον σωστό bitwise τελεστή

```
>>> 1011 __ 1
2022
>>> 2022 __ 1
1011
```

Τελεστές ανά bit (bitwise)

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

- Συμπληρώστε τα κενά με τον σωστό bitwise τελεστή

```
>>> 1011 << 1
2022
>>> 2022 >> 1
1011
```

Τελεστές ανά bit (bitwise)

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

- Συμπληρώστε τα κενά με τον σωστό τελεστέο

```
>>> 0b1010 & ____
2
>>> _____ | 0b101
7
>>> bin(7)
'0b111'
>>> bin(0b1010 ^ _______)
'0b1111'
```

Τελεστές ανά bit (bitwise)

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

- Συμπληρώστε τα κενά με τον σωστό τελεστέο

```
>>> 0b1010 & 0b0111
2
>>> 0b010 | 0b101
7
>>> bin(7)
'0b111'
>>> bin(0b1010 ^ 0b0101)
'0b1111'
```

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

```
>>> x = 13
>>> 2 * x
26
```

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

```
>>> x = 1
>>> x = x + 1
>>> x
2
>>> x = x + 1
>>> x
3
>>> x = x + 1
>>> x
4
```

Εντολή ανάθεσης

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

```
>>> x = '*'
>>> x = _____________
>>> x
'*_*'
>>> x = _____________
>>> x
'*_*_*_*'
>>> x = _____________
>>> x
'*_*_*_*_*_*_*_*'
```

Εντολή ανάθεσης

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

```
>>> x = '*'
>>> x = x + '-' + x
>>> x
'*-*'
>>> x = 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