



ΠΕΡΙΕΧΟΜΕΝΑ:

1. Τριγωνομετρικές Συναρτήσεις
2. Υψώσεις σε Δύναμη και Λογάριθμοι
3. Διασπάσεις Αριθμών και Απόλυτη Τιμή
4. Συνδυαστικοί και άλλοι Υπολογισμοί
5. Υπολογισμοί μεγαλύτερης ακρίβειας

Γιώργος Μ.

Σμαραγδένιος Χορηγός Μαθήματος

Νίκος Θ.

Ασημένιος Χορηγός Μαθήματος

ΜΑΘΗΜΑ 1.1: To module math

1. Τριγωνομετρικές Συναρτήσεις

- Το **module math** περιέχει συνήθεις μαθηματικές συναρτήσεις.
- Είναι ένα «περίβλημα» της βιβλιοθήκης math.h της C επαυξημένο με μερικές ακόμη συναρτήσεις.

Σταθερές:

σταθερά	τιμή
pi	$\pi = 3.141592 \dots$
e	$e = 2.718281 \dots$
tau	$\tau = 6.283185 \dots$
inf	$+\infty$
nan	not a number (NaN)

1. Τριγωνομετρικές Συναρτήσεις

συνάρτηση	επεξήγηση
sin(x)	ημίτονο του x
cos(x)	συνημίτονο του x
tan(x)	εφαπτομένη του x
asin(x)	τόξο ημιτόνου του x
acos(x)	τόξο συνημιτόνου του x
atan(x)	τόξο εφαπτομένης του x
atan2(x, y)	τόξο εφαπτομένης του x/y

Το x είναι σε ακτίνια. Και υπάρχουν και οι εκδοχές sinh, cosh κ.λπ. που υπολογίζουν το υπερβολικό ημίτονο, συνημίτονο κ.λπ.

Παράδειγμα 1:

```
from math import pi, sin, cos, tan
for i in range(4):
    v = i*pi/2
    print(f"sin({i}*Pi/2)={sin(v)}")
    print(f"cos({i}*Pi/2)={cos(v)}")
    print(f"tan({i}*Pi/2)={tan(v)}")
```

Μετατροπές από ακτίνια σε μοίρες και αντίστροφα:

degrees(x)	Μετατρέπει από ακτίνια σε μοίρες
radians(x)	Μετατρέπει από μοίρες σε ακτίνια

Ορίζονται επίσης για τον υπολογισμό αποστάσεων:

συνάρτηση	επεξήγηση
dist(p,q)	Ευκλείδια απόσταση των σημείων p, q Τα p,q είναι ακολουθίες που απεικονίζουν σημεία.
hypot(*coord)	Ευκλείδια νόρμα του σημείου: $\sqrt{x_1^2 + x_2^2 + \dots + x_n^2}$

Άσκηση 1:

Υπολογίστε την απόσταση των σημείων (0,0,0) και (1,2,1) και τη νόρμα του (1,2,1)

Υψώσεις σε Δύναμη

συνάρτηση	επεξήγηση
pow(x,y)	x^y
exp(x)	e^x
sqrt(x)	\sqrt{x}
isqrt(x)	$\lfloor \sqrt{x} \rfloor$
frexp(x)	επιστρέφει $0.5 \leq m \leq 1$ και $e \in \mathbb{Z}$: $x = m \cdot 2^e$
ldexp(x,y)	$x \cdot 2^y$
expm1(x)	$e^x - 1$

Προσοχή:

- Υπάρχει και built-in συνάρτηση pow και ο τελεστής **. Η διαφορά είναι ότι η pow της math κάνει πρώτα μετατροπή των ορισμάτων της σε float.

Παρατήρηση:

- pow(1.0,x)=1.0 και pow(x, 0.0)=1.0
- pow(-float, float): προκαλεί εξαίρεση ValueError
- Η expm1(x) έχει μικρότερο σφάλμα από την exp(x)-1
- sqrt(<0): προκαλεί εξαίρεση ValueError

Άσκηση 2:

Μελετήστε τις επιστρεφόμενες τιμές της frexp για τις διαδοχικές τιμές του x από το 0 έως το 100.

Λογάριθμοι:

συνάρτηση	επεξήγηση
log(x)	$\log_e x = \ln x$
log(x, base)	$\log_{base} x$
log10(x)	$\log_{10} x$
log2(x)	$\log_2 x$
log1p(x)	$\log_e(1 + x)$

Παράδειγμα 2:

```
from math import e, log, log10

print(log(e))

for x in [1,10,100,1000]:
    print(f"log({x}) = {log10(x)}")
```

Άσκηση 3:

Ορίστε μία συνάρτηση που να υπολογίζει τη λογιστική συνάρτηση

$$f(x) = \frac{1}{1 + e^{-x}}$$

Άνω και Κάτω Ακέραιος:

συνάρτηση	επεξήγηση
<code>floor(x)</code>	κάτω ακέραιος: $\lfloor x \rfloor$
<code>ceil(x)</code>	άνω ακέραιος: $\lceil x \rceil$
<code>modf(x)</code>	Επιστρέφει το ακέραιο και το πραγματικό μέρος του x
<code>fmod(x,y)</code>	Επιστρέφει το πραγματικό μέρος της διαίρεσης των πραγματικών x,y

Παράδειγμα 3:

```
from math import floor, ceil

print(floor(5.1), floor(5))
print(ceil(-3.4), ceil(-3))
```

Άσκηση 4:

Πειραματιστείτε με την `modf` (π.χ. εκτυπώστε το ακέραιο και το πραγματικό μέρος του π)

Απόλυτη τιμή:

συνάρτηση	επεξήγηση
<code>fabs(x)</code>	απόλυτη τιμή του x : $ x $
<code>copysign(x,y)</code>	Επιστρέφει την απόλυτη τιμή του x , με το πρόσημο του y

Παρατήρηση:

- Ισοδύναμη λειτουργία έχει και η built-in συνάρτηση `abs(x)`

Άσκηση 5:

Υπολογίστε την απόλυτη τιμή ενός αρνητικού αριθμού, του μηδέν και ενός θετικού αριθμού, τόσο με την `abs` όσο και με την `fabs`

Συνδυαστικοί Υπολογισμοί:

συνάρτηση	επεξήγηση
factorial(n)	$n!$
comb(n,k)	$C(n, k) = \frac{n!}{k! (n - k)!}$
perm(n,k)	$P(n, k) = \frac{n!}{(n - k)!}$

Παρατήρηση:

- n, k πρέπει να είναι ακέραιοι. Αλλιώς προκαλούνται εξαιρέσεις TypeError ή ValueError.

Άσκηση 6:

Μελετήστε τις εξαιρέσεις που προκαλούνται στην comb(n,k) αν εισάγουμε ορίσματα που είναι αρνητικοί ή πραγματικοί και κάνετε κατάλληλο χειρισμό αυτών (εκτυπώσεις αντιστοίχων μηνυμάτων)

Σημαντικές Συναρτήσεις:

συνάρτηση	επεξήγηση
gcd(n,m)	Μέγιστος Κοινός Διαιρέτης των n, m
erf(z)	Error Function του x : $\frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt$
erfc(z)	Complementary Error Function του x : $1 - \text{erf}(x)$
gamma(z)	$\Gamma(z) = \int_0^\infty x^{z-1} e^{-x} dx$
lgamma(z)	$\ln \Gamma(z)$

Άσκηση 7:

Υπολογίστε το $\Gamma(z)$ για $z=2, 3, \dots, 10$ και το $n!$ για $n=1, 2, \dots, 9$

Άθροισμα – Γινόμενο Στοιχείων

συνάρτηση	επεξήγηση
fsum(iter)	Επιστρέφει το άθροισμα των floats του iterable (π.χ. λίστα)
prod(iter)	Επιστρέφει το γινόμενο των στοιχείων του iterable

Παρατήρηση:

- Ισοδύναμη λειτουργία έχει και η built-in συνάρτηση sum(x)

Άσκηση 8:

Εξετάστε τη διαφορά στο αποτέλεσμα μεταξύ της built-in συνάρτησης sum και της fsun αθροίζοντας τα στοιχεία ενός πίνακα ακεραίων.

Άσκηση 9:

Υπολογίστε με την prod το γινόμενο των αριθμών από το 1 έως το 100 σε μία γραμμή κώδικα.

Έλεγχοι:

συνάρτηση	επεξήγηση
isclose(a,b, rel_tol=1e-09, abs_tol=0.0)	Επιστρέφει T/F ανάλογα με το αν τα a,b είναι “κοντά” είτε αν $ a - b \leq abs_tol$ είτε π.χ rel_tol=0.05 (απέχουν 5%)
isinf(x)	True: αν x=inf
isfinite(x)	True: αν x≠inf και x≠ nan
isnan(x)	True: αν x=nan

Παράδειγμα 4: isclose.py

```
from math import isclose

print(isclose(0.001,0.005,abs_tol=1e-3))
print(isclose(0.0001,0.0005,abs_tol=1e-3))
```

Άσκηση 10:

Ελέγξτε αν το nan είναι ίσο με το nan με την ισότητα.