

За роботою зі змінними, мова програмування Python значно відрізняється від багатьох інших мов програмування. Зокрема в Python взагалі не передбачено можливості декларувати змінні, з призначенням їм визначеного типу. Тип змінній приписується через присвоєне значення і може змінюватись протягом програми.

Серед основних числових типів Python: цілочисельні, типи з плаваючою точкою, комплексні, булевий та рядковий.

Цілочисельні типи представлено **int** та **long**.

Тип **int** присвоюється, якщо змінна одержує цілочисельне значення в межах від **-2147483647** до **+2147483647**, що відповідає значенню $\pm 2^{31}-1$. Якщо в розрахунках чи шляхом прямого присвоєння змінна одержує цілочисельне значення, що, за модулем, більше вказаних, то останнє визначається як типу **long**, і після числа виводиться ознака типу - символ **L**. Тип **long** можна приписати до значення явно. Значення типу **long** формально не обмежені.

Тип чисел з плаваючою крапкою (**float**) є аналогом типів **double** в інших мовах програмування - для його зберігання використовують 8 байтів = 64 біти. Тип **float** приписується при присвоєнні числового значення, яке містить десяткову крапку (**3.14**; **2.3**; **.15**; **31.**).

Булевий тип **bool** приписується разом з логічним значенням **True** або **False** (обов'язково з дотриманням написання через велику літеру). З іншого боку, інструкції, де використовуються булеві значення можуть використовувати значення будь-яких типів. Ненульові цілі числа та числа з плаваючою крапкою вважаються рівними **True**, нульові значення вважаються рівними **False**.

Завдання:

1. В режимі програмної оболонки Python Shell виконати наступні дії:

1.1

- присвоїти змінній **a** ціле значення:

```
>>> a=127
```

- переглянути присвоєне значення:

```
>>> a
```

- переглянути тип змінної

```
>>> type(a)
```

Шляхом множення **a** на заданий множник або підняття до заданої степені добитись зміни типу **a** на **long**.

```
>>> a+=1000
```

```
>>> a*=125
```

```
>>> a**=5
```

1.2

З одержаним числом провести зворотні дії, зменшуючи його до меж, допустимих типом **int**.

1.3

- присвоїти змінній *s* значення типу **float**
- провести модифікування значення з постійним контролем типу

1.4

Виконати математичні дії: додавання (+), віднімання (-), множення (*), ділення (/), цілочисельного ділення (//), залишку (%), степені (**) в яких а) обидва аргументи цілі; б) перший аргумент цілий, другий дійсний; в) перший аргумент дійсний, другий цілий; г) обидва аргументи дійсні.

Вибірково повторити дії, використовуючи аргументи булевого типу.

Контролювати тип результату.

2. Скласти, відлагодити та запустити програму, що реалізує виконання п. 1.6 завдання для Pascal, застосувавши усі функції округлення (**ceil**, **floor**, **trunc**) з модуля **math**, а також процедури перетворення (приведення) до типів (**int**, **long**, **float**, **bool**)

Примітка: В завданнях для Pascal, Basic та Python перелік оброблюваних значень доповнити від'ємними значеннями: - **2*pi**, та - **exp(2)**