

Урок 7. Вкладені цикли

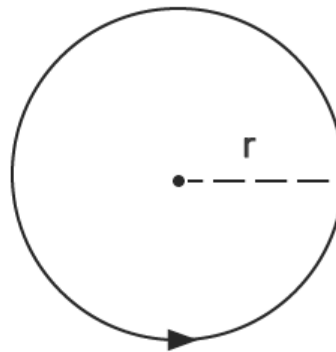
Вивчення нового матеріалу.

Слайд
№1

Розглянемо детальніше команду **circle** побудови кола.

circle(r)

r – радіус кола



Зауваж, що як до, так і після побудови кола черепашка перебуває на тій самій точці кола, причому коло міститься зліва від напрямку, у якому дивиться черепашка.

Слайд
№2

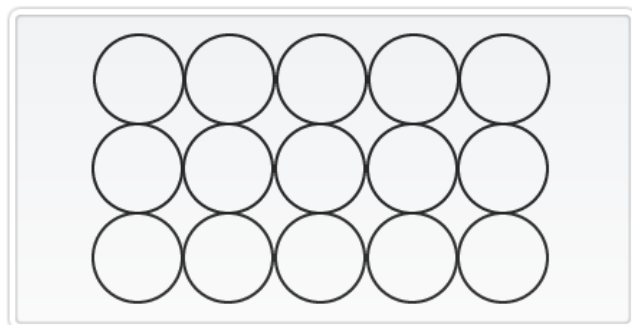
```
import turtle
turtle.shape("turtle")

#перший ряд
for i in range(5):
    turtle.down()
    turtle.circle(30)
    turtle.up()
    turtle.forward(60)

#другий ряд
turtle.goto(0,60)
for i in range(5):
    turtle.down()
    turtle.circle(30)
    turtle.up()
    turtle.forward(60)

#третій ряд
turtle.goto(0,120)
for i in range(5):
    turtle.down()
    turtle.circle(30)
    turtle.up()
    turtle.forward(60)
```

Отже, для побудови усього трьох рядів кіл ми отримаємо доволі громіздку програму. А якщо таких рядів має бути 5 або 105?



Слайд
№3

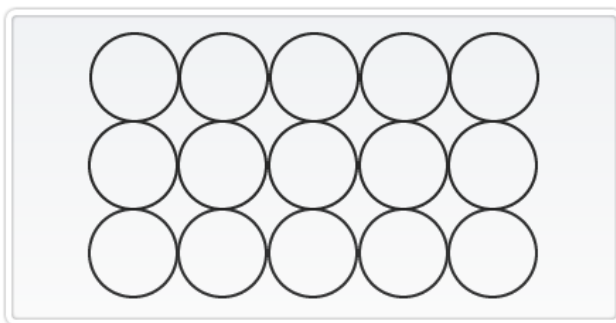
```
import turtle
turtle.shape("turtle")

#перший ряд
turtle.goto(0,0)
for i in range(5):
    turtle.down()
    turtle.circle(30)
    turtle.up()
    turtle.forward(60)

#другий ряд
turtle.goto(0,60)
for i in range(5):
    turtle.down()
    turtle.circle(30)
    turtle.up()
    turtle.forward(60)

#третій ряд
turtle.goto(0,120)
for i in range(5):
    turtle.down()
    turtle.circle(30)
    turtle.up()
    turtle.forward(60)
```

Зробимо невеликі зміни в коді, які не призведуть до жодних змін в зображенні: допишемо команду **goto(0,0)** для першого ряду.



З даної програми видно, що різними є лише значення, які переміщують курсор по осі **y**.

Спробуємо скоротити цю програму.

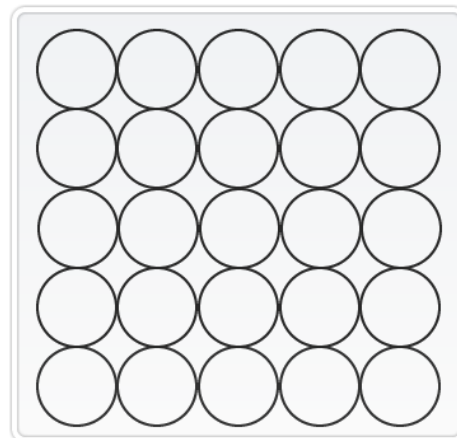
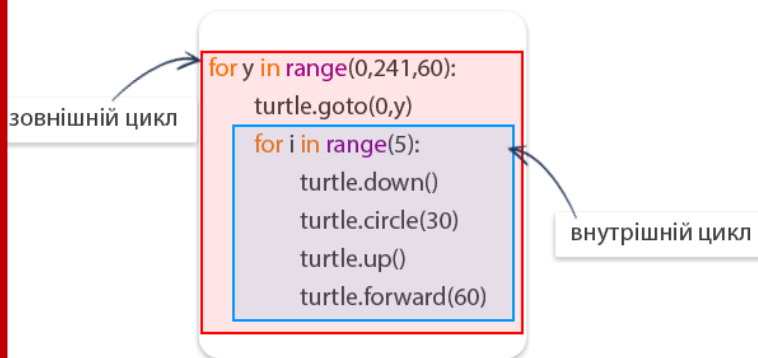
Далі

Слайд
№4

Щоб спростити код, вкладемо один цикл всередину іншого. Лічильником внутрішнього циклу є змінна **i**, що набуває значень **0, 1, 2, 3, 4**, а лічильником зовнішнього циклу буде змінна **y**, що набуватиме значень **0, 60, 120, 180, 240**.

Слайд
№5

Цикли такого типу називають **вкладеними**.



Тут **внутрішній** цикл буде створювати зображення одного ряду кіл, а **зовнішній** — переміщувати курсор на початок нового рядка.



Для спрощення коду ми використали вкладені цикли.

Далі

одиначний відступ

подвійний відступ

одиначний відступ

```
for i in range():  
    for j in range():  
        turtle...  
        turtle...  
        turtle...  
        ...  
    turtle...
```

тіло внутрішнього
циклу

команда в тілі зовнішнього,
але не внутрішнього циклу



Що куди вкладено, у Python визначається відступами.

Далі

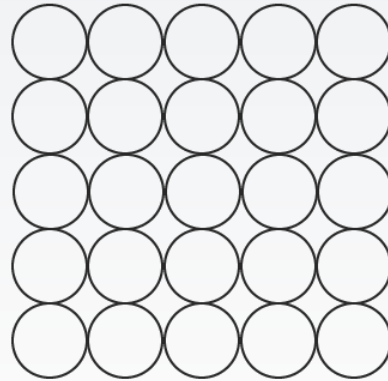
Вправа 1

Вправа №1



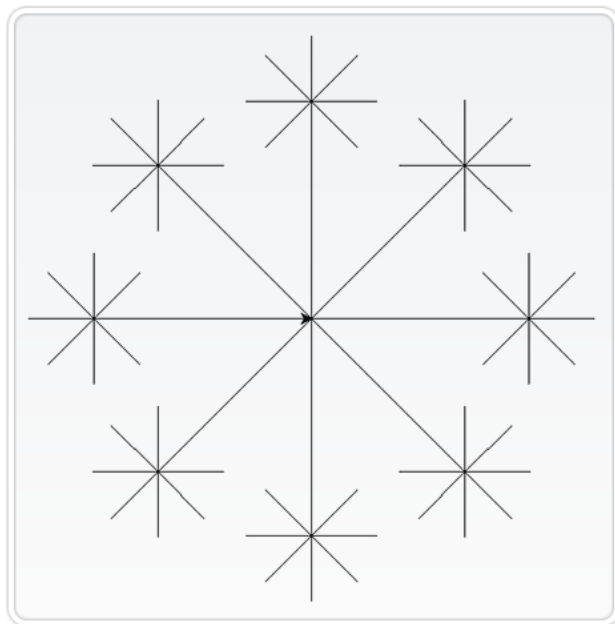
Вправа 1. Створи програму в середовищі Python.

Створи програму побудови
зображення згідно зразка.



Слайд
№7

Тепер розглянемо побудову
ось такої сніжинки.



Створимо ще одну програму із вкладеними циклами.

Далі

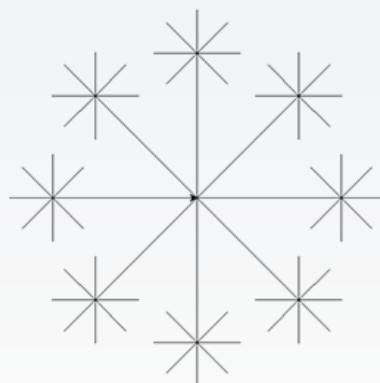
Вправа 2.

Вправа
№2



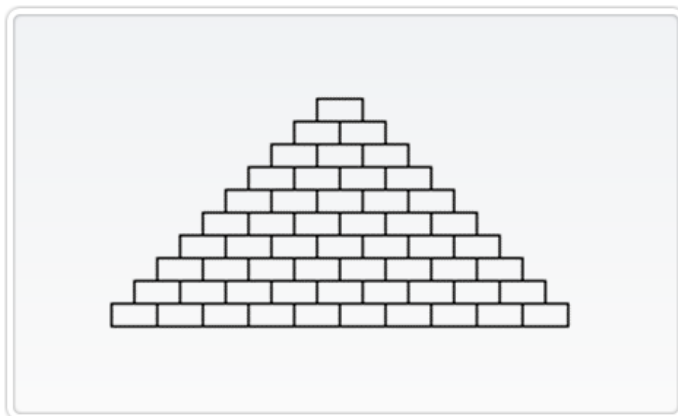
Вправа 2. Створи програму в середовищі Python.

Створи програму побудови
зображення сніжинки



Слайд
№8

Тепер ми спробуємо побудувати пірамідку з прямокутних цеглин.

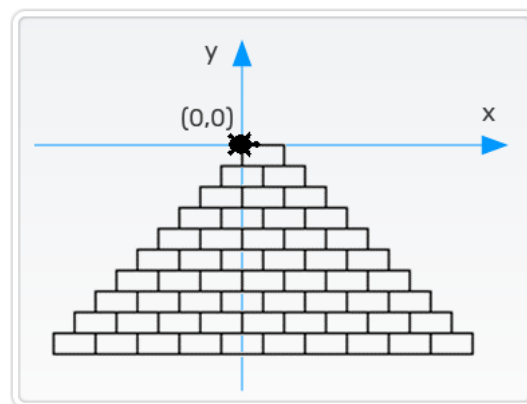


Створимо програму, що малює пірамідку. Тут теж потрібні вкладені цикли!

Слайд
№9

Піраміду малюватимемо зверху вниз. Тоді після побудови кожного рядка кількість цеглин наступного рядка збільшуватиметься на 1.

З рисунка видно, що якщо початкове положення черепашки $(0,0)$ зіставити з верхнім лівим кутом найвищої цеглини піраміди, то координати початку наступних рядів будуть від'ємними.



Зверни увагу на порядок побудови рядів піраміди.

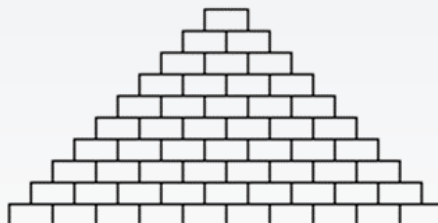
Далі

Вправа
№3



Вправа 3. Створи програму в середовищі Python.

Створи програму побудови
піраміди з цеглин.



Створи програму побудови пірамідки в
середовищі Python та перевір правильність її виконання.

Готово