



Become QA Auto

Лінійні алгоритми (задача 1)

Бутенко Сергій

```
le.py > ...
x = "PiMyLifeUp" #First assign the 'x' variable a string
type(x) #Output the data type of our 'x' variable
# 43 #Now assign it a integer value
type(x) #Output its type again

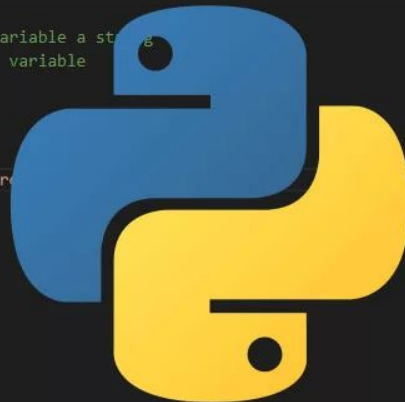
days = 5
int("There are " + str(days) + " days r

b, c = "Apple", "Banana", "Cherry"

b = c = "Lime"

= "Hello World"
= 453
int(x)
int(y)

= "Life Up"
```





План лекції



Задача 1.

Визначити довжину огорожі для прямокутної ділянки



Постановка задачі



Побудова алгоритму у вигляді блок-схеми



Побудова математичної моделі



Протокол виконання алгоритму



Задача 1

Скласти програму, яка буде визначати довжину огорожі для прямокутної ділянки

Постановка задачі

Аргументи:

ширина і довжина ділянки.

Шуканий результат:

довжина огорожі

Математична модель

Модель:

прямокутник.

Допустимі дані: додатні цілі числа.

Очікуваний результат:

додатне ціле число

Зв'язок між аргументами і результатом:

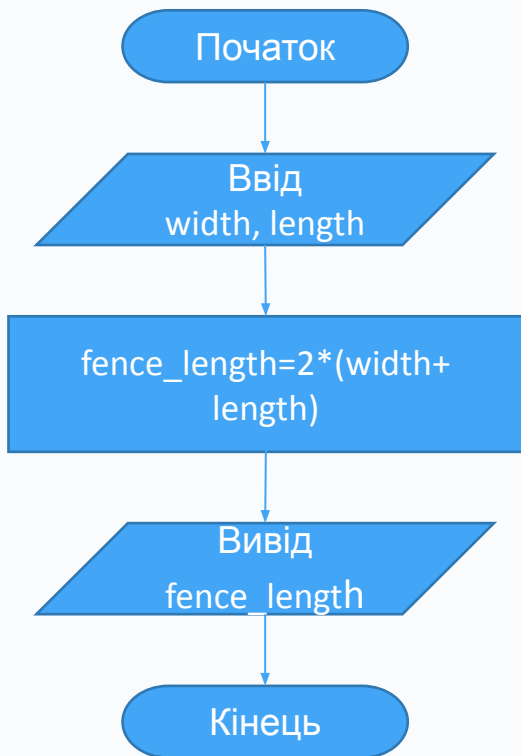
Формула периметра
прямокутника $P=2(a+b)$

Алгоритм

1. Початок
2. Ввести довжину і ширину `width, length`
3. Обчислити
`fence_length=2*(width+length)`
4. Вивести результат
`fence_length`
5. Кінець



Блок-схема алгоритму



Початок – комп'ютер розпочинає виконання алгоритму

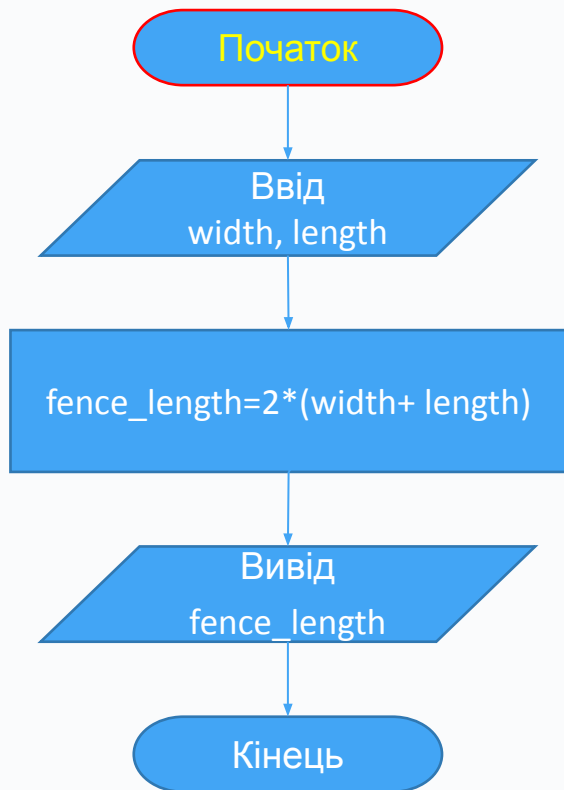
Аргументам **width**, **length** присвоїти значення, введені користувачем з клавіатури

Обчислення: змінній **fence length** присвоїти результат обчислення виразу **2 * (width + length)**

Вивід **fence length** - вивести знайдене значення периметру як результат виконання алгоритму

Кінець – комп'ютер завершує виконання алгоритму

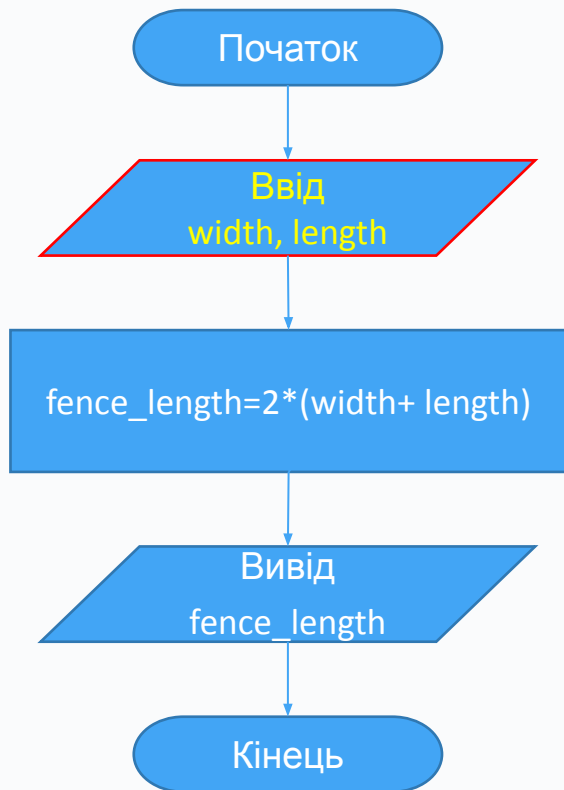
Протокол виконання алгоритму



Вміст пам'яті

width	length	fence_length

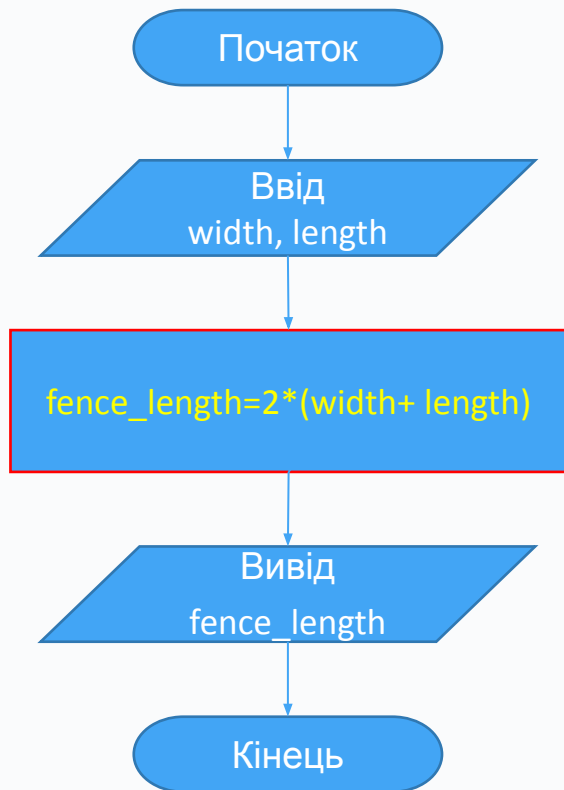
Протокол виконання алгоритму



Вміст пам'яті

width	length	fence_length
12	25	

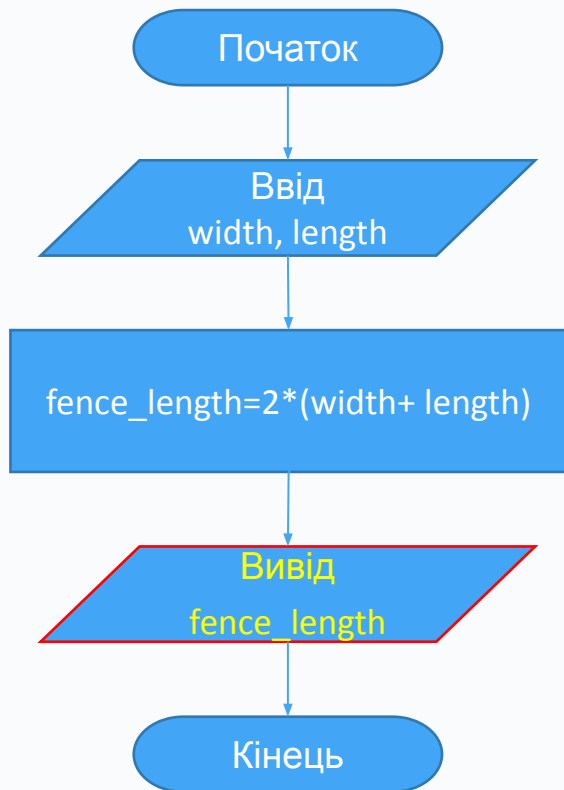
Протокол виконання алгоритму



Вміст пам'яті

width	length	fence_length
12	25	74

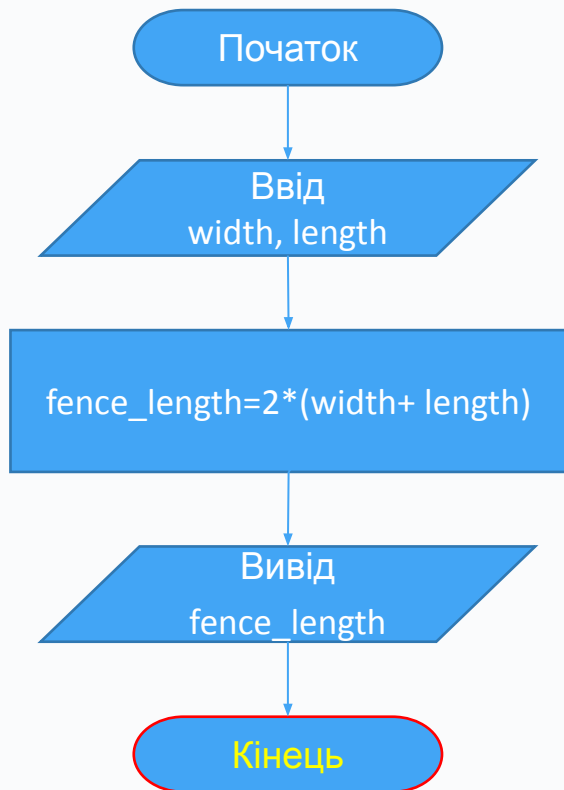
⦿ Протокол виконання алгоритму



Вміст пам'яті

width	length	fence_length
12	25	74

Протокол виконання алгоритму

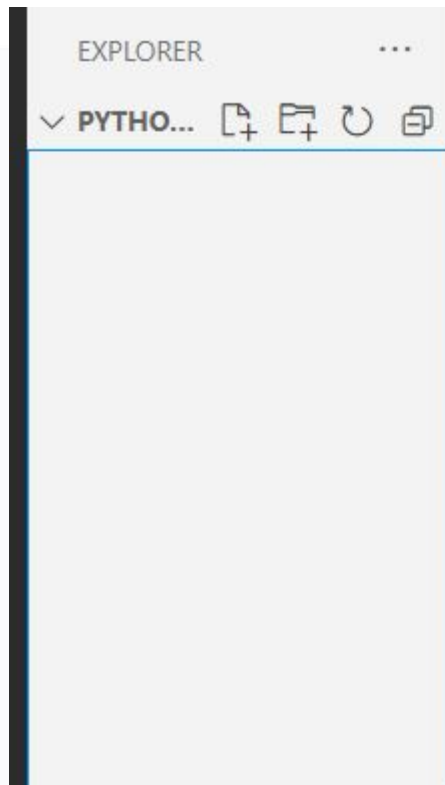
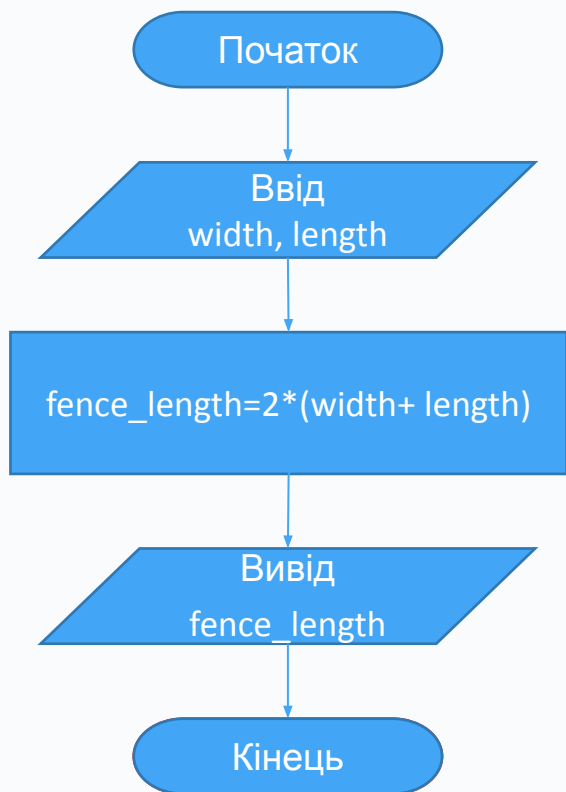


Вміст пам'яті

width	length	fence_length
12	25	74

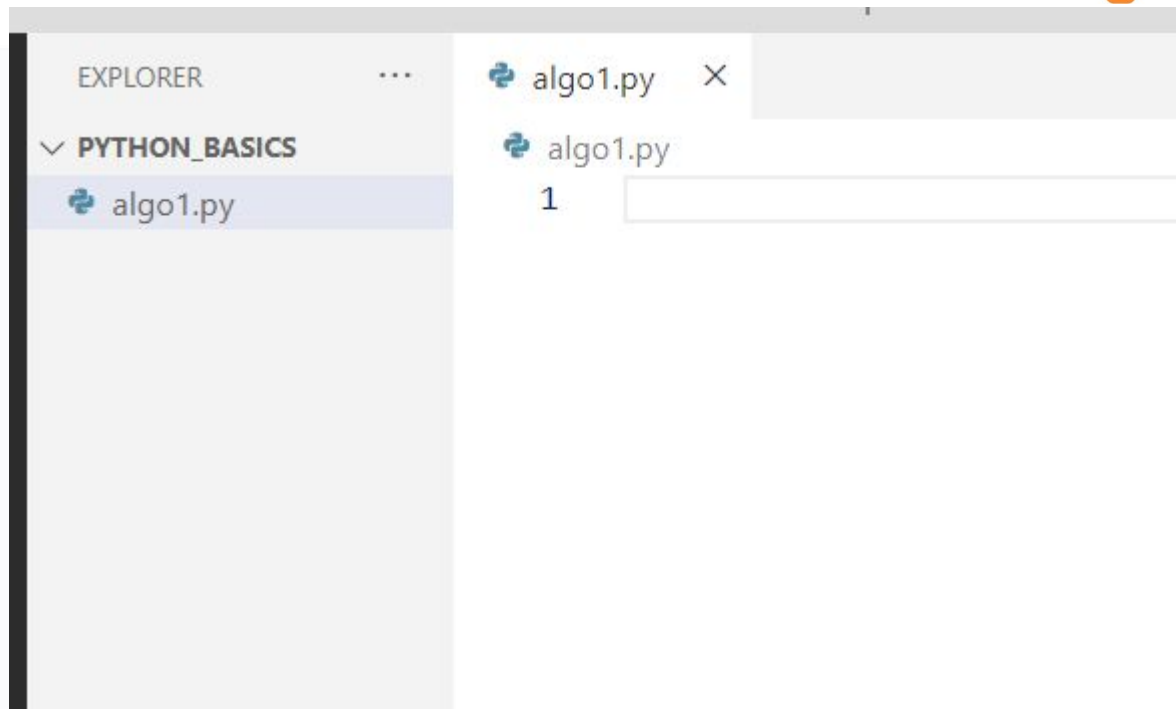
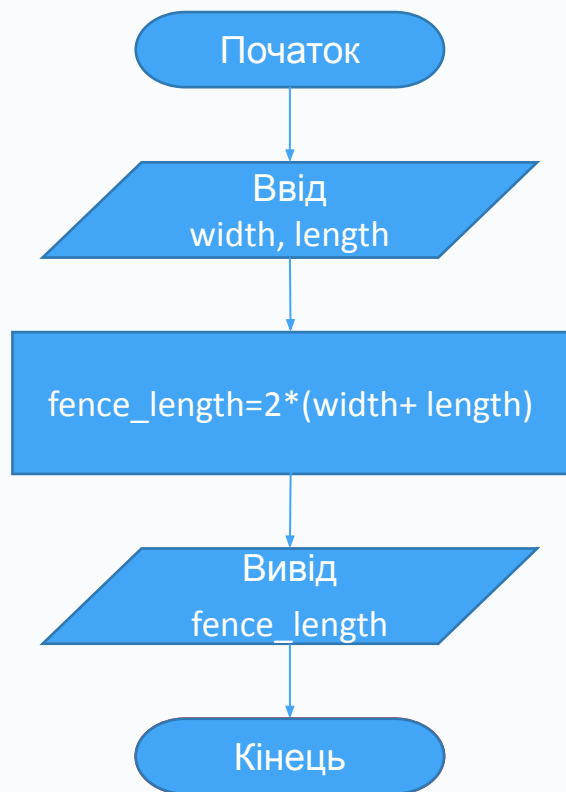


Практика



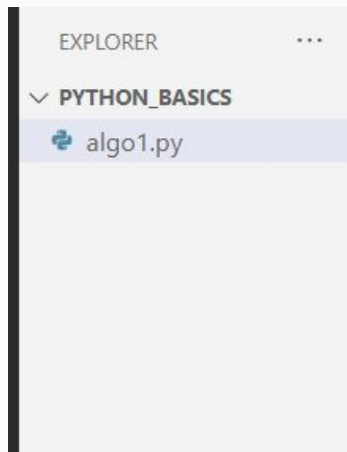
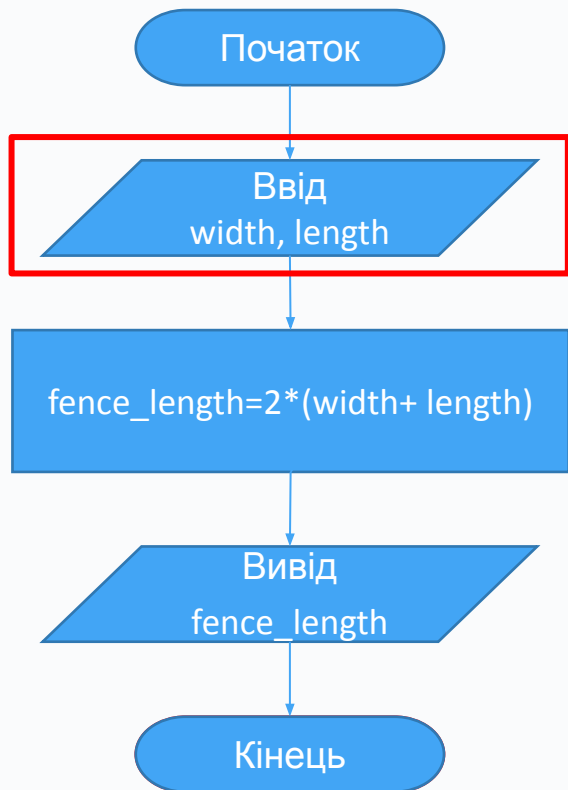


Практика





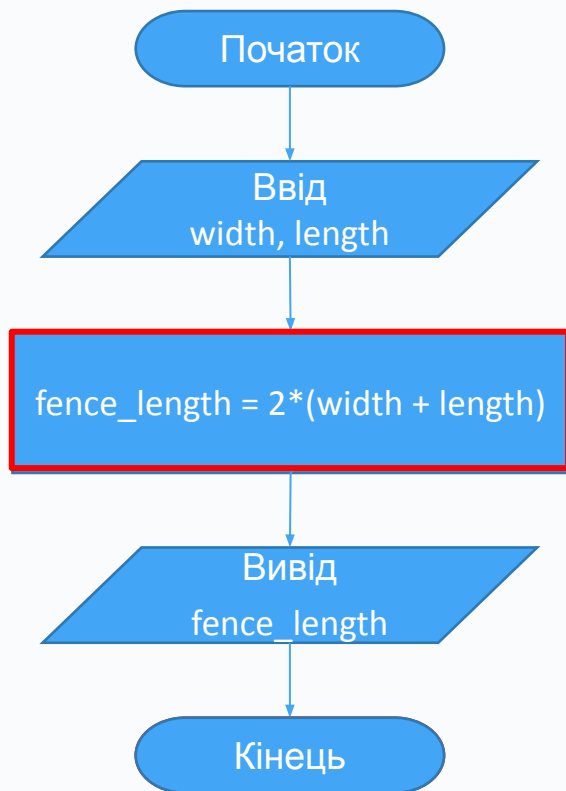
Практика



```
algo1.py  X
algo1.py > ...
1 width = int(input("Введіть ширину: "))
2 length = int(input("Введіть довжину: "))
3 |
```



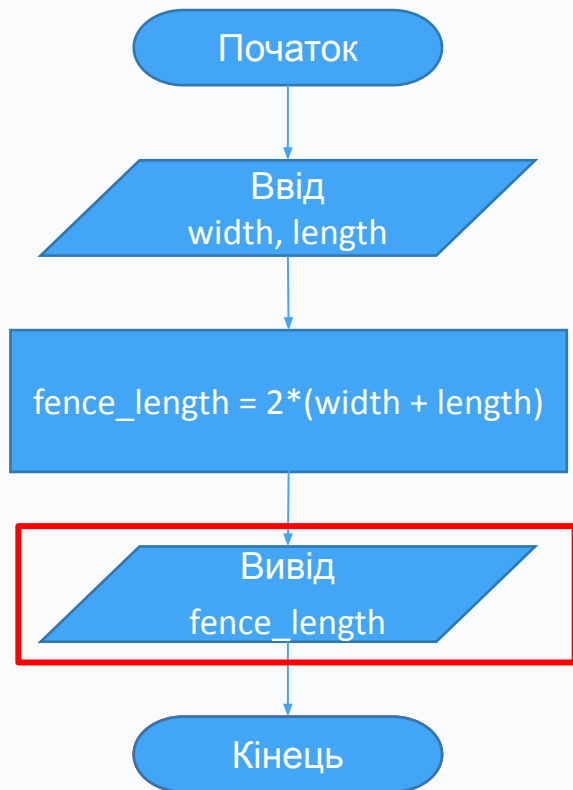
Практика



```
EXPLORER  ...
└─ PYTHON_BASIC
    └─ algo1.py

algo1.py
1 width = int(input("Введіть ширину: "))
2 length = int(input("Введіть довжину: "))
3
4 fence_length = 2 * (width + length)
```

Практика



algo1.py > ...

```
1 width = int(input("Введіть ширину: "))
2 length = int(input("Введіть довжину: "))
3
4 fence_length = 2 * (width + length)
5 print("Периметр =", fence_length)
```



Практика



algo1.py X

algo1.py > ...

```
1 width = int(input("Введіть ширину: "))
2 length = int(input("Введіть довжину: "))
3
4 fence_length = 2 * (width + length)
5 print("Периметр =", fence_length)
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

JUPYTER

```
PS C:\Users\sergii.butenko\python_basics> & C:/Users/sergii.butenko/python_basics/alg
on_basics/alg
Введіть ширину: 2
Введіть довжину: 3
Периметр = 10
PS C:\Users\sergii.butenko\python_basics> █
```



Підсумки



Побудували блок-схему алгоритму обчислення довжини огорожі за відомими розмірами ділянки



Дослідили роботу алгоритму за допомогою протоколу його виконання