

## Тема. Алгоритми впорядкування масиву (списку)

### Очікувані результати заняття

Після цього заняття треба вміти:

описувати алгоритми впорядкування масиву, сортувати список різними способами, тестувати створений код

### Повторюємо

- які способи заповнення масиву ви знаєте?
- як знайти елемент списку, що відповідає певній умові?
- як знайти середнє арифметичне елементів масиву?

### Ознайомтеся з інформацією

## Способи сортування

#### Обміном

Попарно перевіряючи елементи, при потребі міняємо їх місцями

#### Вставкою

Список ділиться на 2 частини: відсортовану та невідсортовану. Береться перший невідсортований елемент і шукається місце для нього

#### Вибором

Шукаємо найменший елемент і ставимо його на першу позицію, наступний найменший – на другу...

#### Злиттям

Формування нового впорядкованого списку з двох впорядкованих

### Перегляньте відео

Сортування вибором: [https://youtu.be/PJno7U\\_sfdc](https://youtu.be/PJno7U_sfdc)

Сортування вставкою: <https://youtu.be/RlTXEXPxVpk>

Сортування обміном (бульбашкою): <https://youtu.be/kamA9pbpH-o>

### Виконайте у середовищі програмування

1. <https://repl.it/languages/python3>

або

2. <https://www.techage.in/online-python-compiler/>

## Задача 1. Сортуння обміном. Метод "бульбашки"

Проходимо список стільки разів, скільки у ньому є елементів. Щоразу переглядаємо увесь список (крім останнього елемента) (можна до кінця відсортованої частини). При потребі міняємо елементи місцями.

```
import random
spysok=[]
for nomer in range(10):
    k=random.randint(0,5)
    spysok.append(k)
print(spysok)
for perebir in range(9):
    for nomer in range(9):
        if spysok[nomer]>spysok[nomer+1]:
            k=spysok[nomer]
            spysok[nomer]=spysok[nomer+1]
            spysok[nomer+1]=k
    print(spysok)
```

## Задача 2. Сортуння вибором

Проходимо список стільки разів, скільки елементів. Щоразу знаходимо найменший елемент та його номер і міняємо його місцями з поточним

```
for perebir in range(9):
    print(spysok[perebir:10])
    fragment=spysok[perebir:10]
    m=fragment.index(min(fragment))
    k=m+perebir
    spysok[k], spysok[perebir]=spysok[perebir], spysok[k]
print(spysok)
```

## Інший запис (без функцій min та index)

```
for n in range(10):
    m=spysok[n]
    for nomer in range(n,10):
        if m>=spysok[nomer]:
            m=spysok[nomer]
            a=nomer
    k=spysok[n]
    spysok[n]=spysok[a]
    spysok[a]=k
print(spysok)
```

## Задача 3. Сортуння вставкою

Проходимо список стільки разів, скільки елементів, починаючи з другого. Записуємо елемент у допоміжну змінну, і проходимо список у зворотньому напрямку, поки не знайдемо правильне місце для елемента. При цьому всі пройдені елементи зміщуються на один.

```
for perebir in range(10):
    j = perebir - 1
    element = spysok[perebir]
    del spysok[perebir]
    while (j >= 0) and (spysok[j] > element):
        j = j-1
    spysok.insert(j + 1, element)
print(spysok)
```

## Інший спосіб (без функцій del та insert)

```
for n in range(1,10):  
    a=n  
    print(spysok[a])  
    while (spysok[a] > spysok[a-1]) and (a > =0):  
        k=spysok[a]  
        spysok[a]=spysok[a-1]  
        spysok[a-1]=k  
        a-=1  
    print(spysok)
```

## Для тих, хто хоче знати більше

<https://dystosvita.org.ua/mod/page/view.php?id=518>

## Домашнє завдання

- Вивчити алгоритми сортування списку
- Написати та реалізувати код до задачі

**Задача 4.** Відсортувати будь-яким способом список з 15 довільних чисел та знайти суму цих чисел

Скріншот робочого коду для розв'язування задачі надіслати на HUMAN або на електронну пошту [nataliartemiuk.55@gmail.com](mailto:nataliartemiuk.55@gmail.com)

## Джерела

- <https://dystosvita.org.ua/mod/page/view.php?id=518>
- <https://panasjukirina.wordpress.com>