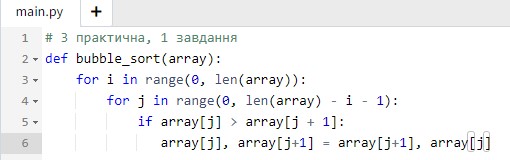
**Практична робота № 3**

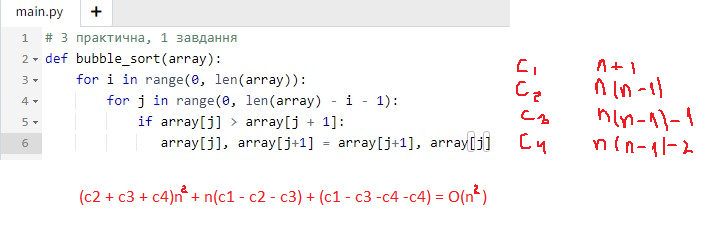
**Тема**. Алгоритми сортування та їх складність. Порівняння алгоритмів сортування   
**Мета:** опанувати основні алгоритми сортування та навчитись методам аналізу їх асимптотичної складності

**Хід роботи**

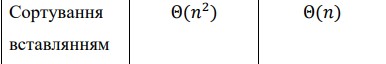
***1.*** *Вивчити самостійно і записати (будь-яким способом) алгоритм бульбашкового сортування*



*Оцінити асимптотику алгоритму сортування методом бульбашки в найгіршому і в найкращому випадку (вибачаюсь, не дуже точно зрозумів спосіб оцінки алгоритмів)*

****

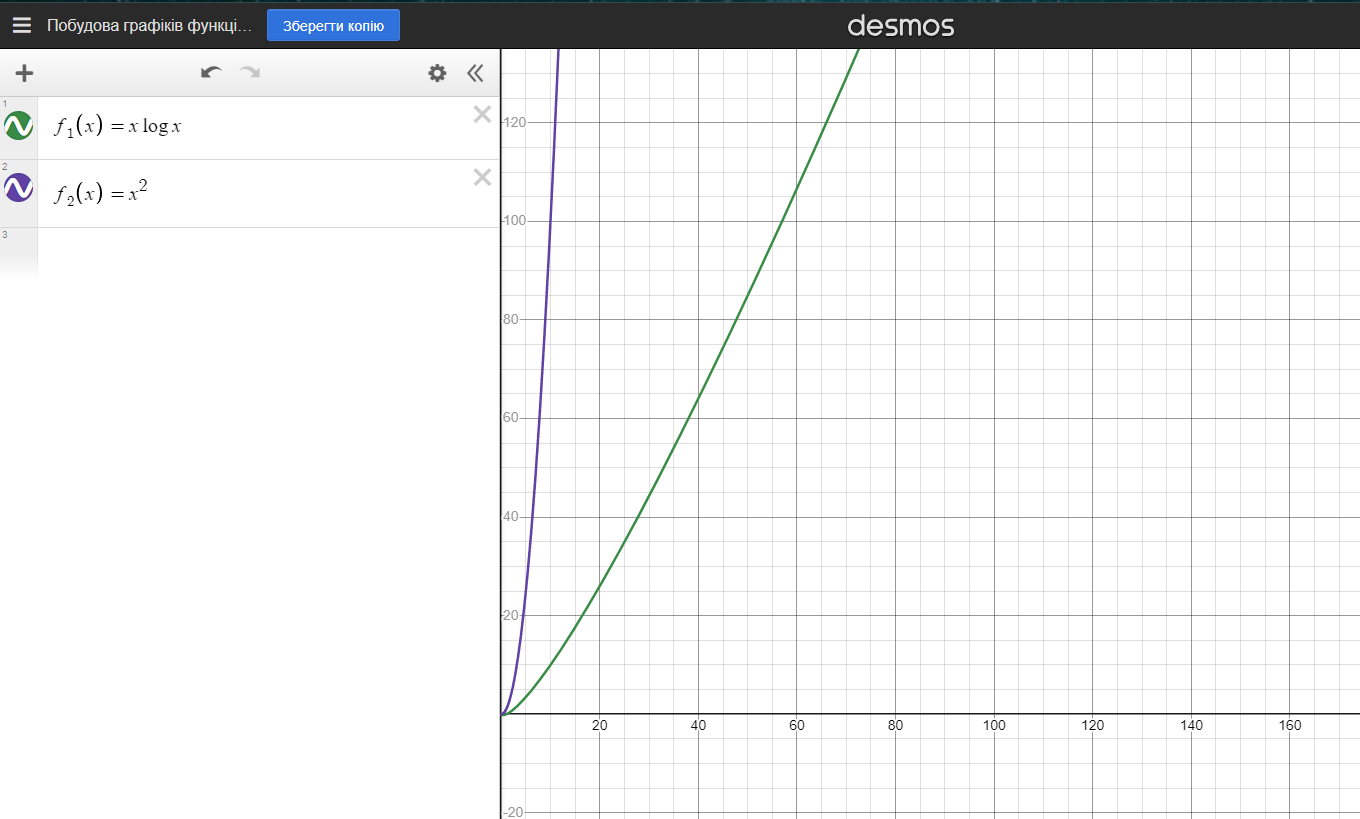
*Порівняти за цими показниками бульбашковий алгоритм з алгоритмом сортування вставлянням*

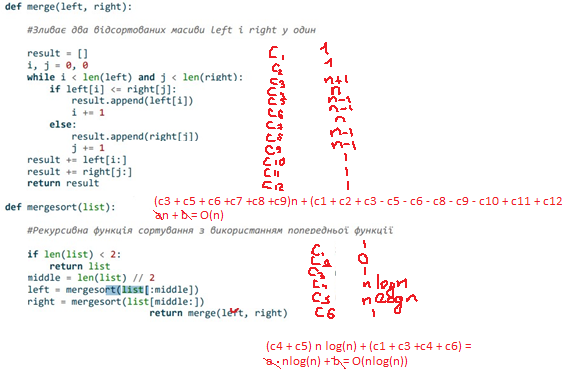


Як бачимо алгоритм вставлянням більш ефективний для найкращого випадку

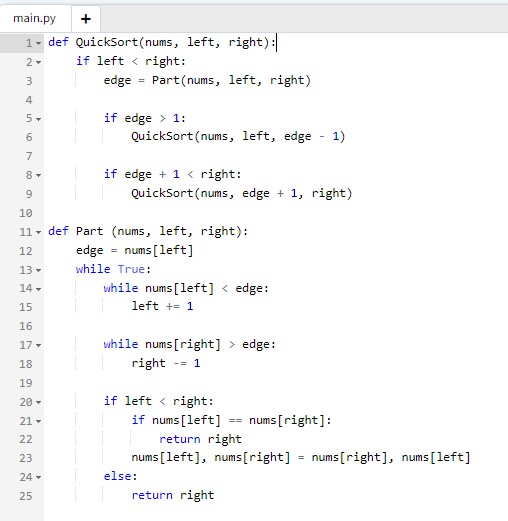
(на фото найкращий справа)

*Чому на практиці бульбашковий алгоритм виявляється менш ефективним у порівнянні з сортуванням методом зливанням?*  
Тому що алгоритм зливання має асимптотичну складність nlog(n), як бачимо по графіку, є величезна різниця у кількості операцій для великих n

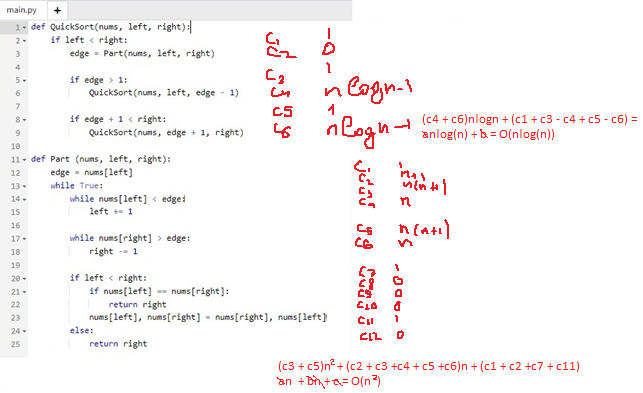


***2.*** *Оцінити асимптотичну складність алгоритму сортування зливанням, скориставшись основною теоремою рекурсії*

***3.*** *Вивчити і записати (будь-яким способом) самостійно алгоритм швидкого сортування*



*Оцінити асимптотичну складність алгоритму швидкого сортування, скориставшись основною теоремою рекурсії*



**Контрольні питання**

*Що таке асимптотична складність алгоритму сортування і чому вона важлива для порівняння алгоритмів?*

Асимптотична складність алгоритму - це спосіб оцінки ефективності алгоритму.  
Показує примірну кількість операцій, в даному випадку, для сортування масиву.  
Від кількості операцій залежить і час, який потрібний для повного сортування.

А асимптотична складність дозволяє визначити, який алгоритм буде найефективнішим для нашого набору даних.

*Які алгоритми сортування мають квадратичну складність у найгіршому випадку? Поясніть, чому це може бути проблемою для великих обсягів даних*

Із тих алгоритмів, що ми вивчили – це сортування бульбашкою, вибором  
вставлянням і швидке сортування. Для великого набору даних, наприклад,  
10 000 чисел в масиві, буде виконано 100 000 000 операцій, а якщо асимптотична складність буде log(n), це буде всього 100 операцій.

*В чому полягає перевага сортування злиттям над сортуванням вставками для великих наборів даних?*

У ефективності, сортування злиттям у найгіршому випадку матиме асимптотичну складність nlog (n), коли вставлянням

*Які алгоритми сортування використовуються для сортування списків у стандартних бібліотеках мов програмування, таких як Python, Java або C++?*

В інтернеті сказано що вони мають якусь свою систему сортування «Timsort».

У логіку є не поглиблювався, але візуально схоже на швидке сортування.

*Яка різниця між алгоритмами сортування злиттям і швидким сортуванням? У яких випадках краще використовувати кожен з цих алгоритмів?*

Знову ж таки у ефективності, сортування злиттям є більш ефективним  
у найгіршому випадку, хоча є однаковими у найліпшому,  
але сортування злиттям виконується не на місці, у відмінності від   
швидкого сортування.

*Які фактори слід враховувати при виборі алгоритму сортування для конкретної задачі?*

Кількість вхідних даних, місце виконання, стан вхідних даних(рівень відсортованості)