

ОС і СП

Лабораторна робота 9: графічні операції

[В папках "DemoGraphEdit for lab8,9,10" і "Graph operations" є демонстраційні та інші матеріали, які можна використати для вивчення і виконання роботи.]

Ця робота є продовженням лабораторної роботи 8. Для виконання даної лабораторної роботи треба мати основу виконання базових графічних операцій, а саме – малювання фігур і збереження та відновлення з файла.

Завдання

Скласти алгоритми і виконати програмну реалізацію окремих графічних операцій за переліком, записаним в пунктах 1, 2, 3 нижче. Можна додати інші свої визначені операції.

1. Вдосконалити раніше реалізовані операції малювання фігур за рухом мишки: додати можливість скасування операції **до її закінчення** шляхом натискання на клавішу Esc:

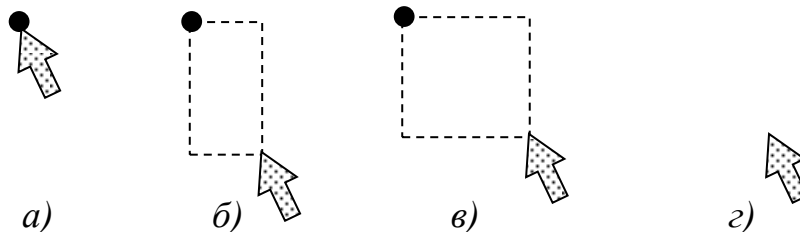


Рис. 1. Малювання стандартної фігури з скасуванням

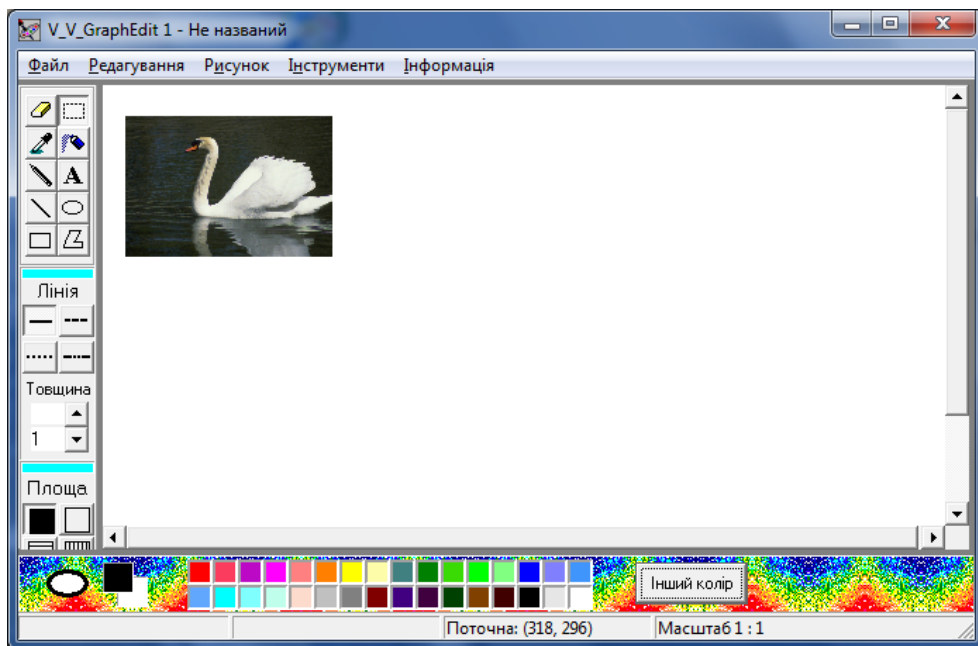
Початок фігури фіксується у момент натискання на клавішу мишки (рис. 1,а). При пересуванні мишки з натисненою клавішею малюємо фігуру від точки початку фігури до точки поточного розташування мишки (рис. 1,б). При подальшому пересуванні мишки стираємо попередню фігуру і малюємо її в новому місці (рис. 1,в) – така операція виконується багатократно, при кожній зміні позиції мишки. При натисненні на клавішу Esc стираємо фігуру (рис. 1,г) і скасовуємо операцію.

2. Додати функціональну можливість скасувати вже **виконані і зафіксовані** операції. Для цього, по-перше, треба додати пункт меню (чи кнопку) скасування останньої операції. По-друге, треба зберігати історію малювання фігур, тобто порядок виконаних графічних операцій і їх параметри. Історія малювання залежить від методу реалізації графічних операцій: векторна графіка чи растрова графіка. Для векторної графіки можна, наприклад, просто

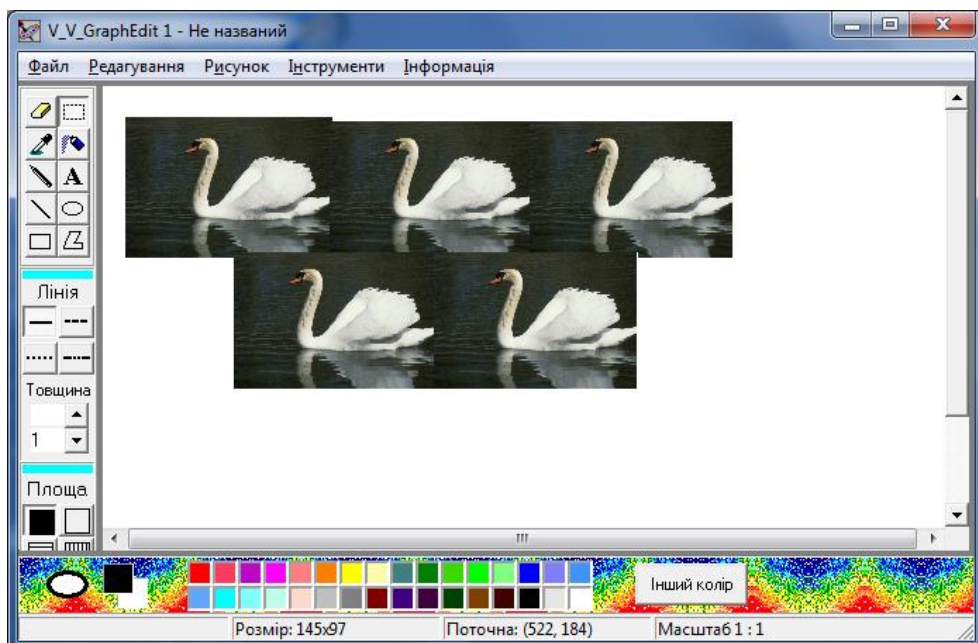
зберігати фігури, параметри і операції (команди малювання) в загальному списку, редагувати список і виконувати перемальовування за цілим списком. Для растрової графіки можна перед кожною операцією робити копію ділянки малюнка, яка буде змінена, чи копію цілого малюнка. Такий варіант в цілому є неоптимальний і може вимагати великого обсягу пам'яті, бо растрова графіка зберігається як масив кольорів кожного пікселя (а який спосіб оптимізації ви можете запропонувати?).

3. Додати можливість копіювання і пересування за рухом мишки окремих фігур чи ділянок малюнка, наприклад:

а) отримали деяке зображення:



б) виконали операції копіювання і пересування ділянок:



Варіант реалізації.

Для цього треба додати в меню або до кнопок команду "Виокремити". Реалізація цієї команди подібна до команд малювання стандартних фігур, лише виконуємо не малювання, а зображення контура виокремленої ділянки. Після виокремлення командою Сору зберігаємо в пам'яті виокремлену ділянку. Тепер виконуємо декілька разів команду Paste і пересуваємо щоразу нову копію в потрібне місце.

N. Запропонуйте інші графічні операції, запишіть алгоритми їх виконання і виконайте програмну реалізацію. В такому разі можна отримати додаткові бали за роботу.

Звіт за роботу

В результаті виконання завдання надіслати:

- 1) архів проєкту (чи окремі файли - для Python);
- 2) скріншоти виконання тестових прикладів на власному комп'ютері - окремо, НЕ АРХІВОВАНІ ФАЙЛИ, або додані до файлу звіту; такі скріншоти є важливі, бо комп'ютерна графіка може мати особливості залежно від поточних налаштувань пристроїв;
- 3) письмовий файл звіту за лабораторну роботу (якщо буде) надсилайте окремим від проєкту (архіву) НЕ АРХІВОВАНИМ файлом формату pdf чи doc.