

## Деякі пояснення виконання операцій навчального графічного редактора V\_V\_GraphEdit

- 1) Основна панель вибору типових *команд* малювання є вгорі ліворуч вікна. Під нею є панелі вибору *параметрів* малювання. Отже, є окремо *команди* і є окремо *параметри*.
- 2) Команда меню "Інструменти → Панелі" керує наявністю чи відсутністю відповідної групи інструментальних кнопок для вибору *параметрів* малювання і просто демонструє можливість керування панелями.
- 3) Кольори для малювання можна обирати з палітри внизу вікна *лівою* клавішею мишки – для ліній і контурів фігур, *правою* клавішею – для зафарбування площини фігур чи об'єктів.
- 4) Натисканням на клавішу Esc або пробілу можна скасувати операцію в процесі виконання *до її закінчення* – поки не відпущена клавіша мишки чи поки не завершені всі кроки операції. Скасування (відміну) операції після фіксації результату не передбачено в цій версії графічного редактора.
- 5) Після виконання операції вставки з буфера чи з файлу зберігається виокремлення вставленого елемента і його можна пересунути мишкою до потрібної позиції. Щоб зафіксувати остаточну позицію – натиснути мишкою де-небудь збоку елемента чи обрати до виконання іншу команду, але не параметри. Щоб скасувати – натиснути на клавішу Esc або пробілу.
- 6) Операція фільтрування "Рисунок → Фільтр" застосовується до виокремленого прямокутного фрагмента малюнка. Якщо виокремлення немає – тоді до цілого малюнка. Фільтрування має демонстраційний характер, на практиці застосовують більше фільтрів і швидкі способи виконання фільтрування.
- 7) В цій версії графічного редактора рекомендується визначити ширину і висоту полотна малювання на самому початку – командою "Рисунок → Атрибути". Частина рисунка, яка виходить за межі полотна, обрізається.

---

### Алгоритми реалізації окремих команд

#### Малювання стандартних фігур

Розглянемо кроки малювання стандартних фігур на прикладі прямокутника. Початок фігури фіксується у момент натискання на клавішу мишки (рис.1, *а*). При пересуванні мишки з натисненою клавішею малюємо прямокутник від точки початку фігури до точки поточного розташування мишки (рис.1, *б*). При подальшому пересуванні мишки стираємо попередній прямокутник і малюємо його у новому місці (рис.1, *в*) – така операція виконується багаторазово, при кожній зміні позиції мишки. Остаточний прямокутник фіксується у момент відпускання клавіші мишки (рис.1, *г*).

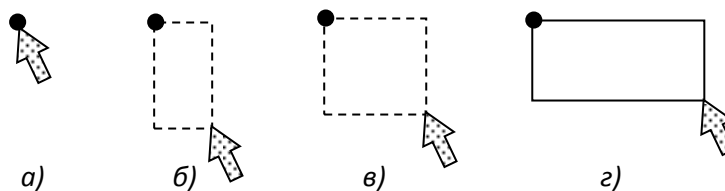


Рис. 1. Кроки малювання стандартної фігури

### Особливості малювання багатокутника

У загальному випадку замкнений багатокутник (в іншій термінології – *полігон*) можна малювати описаним вище методом для прикладу прямокутника. Однак усі проміжні полігони до остаточного вибору не показують перспективи продовження шляхом додавання нових ланок і дають неправильну уяву про кінцевий результат. Тому варто у процесі побудови малювати лише окремі ребра, які вже вибрано (Polyline). Необхідно домогтися фіксування вершин полігона. З цією метою можна використати подію OnClick. Наприклад, на рис.2, а зазначено вже побудовану частину полігона. Наступне ребро для продовження полігона малюємо подібно, як звичайну лінію, описаним вище методом (рис.2, б, в), лише пересувати мишку можна й без натисненої клавіші. Наступне ребро фіксується подальшим натисненням клавіші мишки (подія OnClick) (рис.2, г). Закінчення побудови полігона – за подією OnDblClick, при цьому остання вершина сполучається з найпершою (рис.2, д).

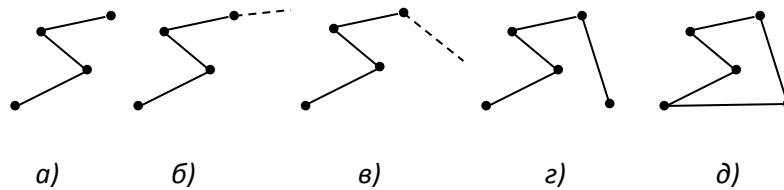


Рис. 2. Кроки малювання полігона

### Стирання за допомогою функції “гумка”

Стирання – це зафарбування ділянки білим кольором. Білий колір для стирання обрано з тих міркувань, що початкове полотно (тло) має білий колір, однак за необхідності колір для стирання можна змінити. Стирання виконується для прямокутної ділянки, розміри якої збігаються з розмірами зображення “гумки” – курсора для команди стирання. Стирання відбувається: за натисненої клавіші мишки (рис.3, а); при пересуванні мишки з натисненою клавішею (рис.3, б). Існують два різні способи стирання. За першим способом (рис.3, в) замальовується лише прямокутна ділянка в місці розташування курсора мишки в момент виникнення події OnMouseMove. Якщо мишку пересувати швидко, то стирання буде окремими “плямами”, як на рис.3, б. Однак при повільному пересуванні мишки можна отримати ефект замальовування суцільної смуги. За другим способом (рис.3, г) замальовується не тільки прямокутна ділянка в місці розташування курсора мишки, але й “коридор” між двома сусідніми позиціями мишки, незалежно від швидкості руху мишки. Отже, стирання завжди відбуватиметься суцільною смугою. Для правильного визначення ширини “коридору” та його координат необхідно враховувати напрям пересування мишки (рис.3, г, д), вибираючи за базу головну діагональ зображення курсора (рис.3, г) або допоміжну (рис.3, д). Очевидно, що необхідно завжди пам’ятати попередню позицію курсора за подією OnMouseMove.

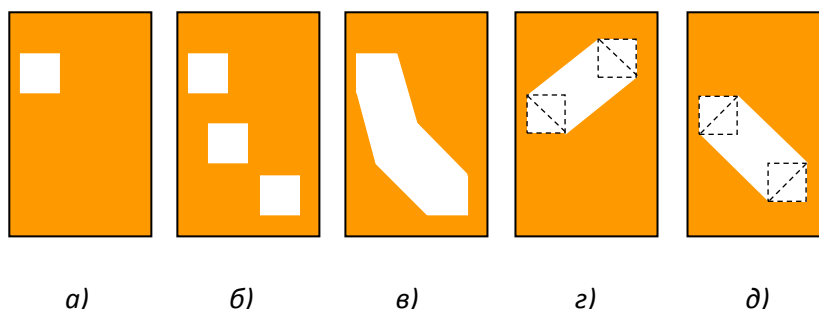


Рис. 3. Стирання малюнка

### Друкування текстових підписів

Ця операція є найбільш громіздкою з-поміж інших і виконується у декілька кроків: 1) вибрати команду формування текстового підпису кнопкою панелі інструментів; 2) показати вікно (форму) з параметрами шрифту FormFontAttr (рис.4); вікно можна пересувати у зручну позицію площини малюнка і закривати, проте не можна змінювати його розміри; позиція вікна визначається параметрами LastPosX, LastPosY, і наступного разу вікно з'явиться в останній збереженій позиції [LastPosX, LastPosY]; 3) виокремити мишкою ділянку у вигляді контурного прямокутника функцією DrawFrameRegion, малювання прямокутника відбувається функцією DrawFocusRect; малювання можна припинити клавішами Esc або пропуску; текст буде розташований у межах виокремленої ділянки; рядки, які не помістяться, будуть відкинуті цілковито або частково; 4) перевести фокус уведення до виокремленої ділянки і ввести з клавіатури необхідний текст, у тім числі розділові знаки, пропуски, перехід на новий рядок за клавішею Enter; якщо рядків буде більше, ніж дозволяє висота ділянки, візуально спостерігається прокручування тексту вгору у межах ділянки, однак зафіксовані будуть лише перші рядки до нижньої межі ділянки, тому не варто створювати ситуацію прокручування тексту – це спотворює сприйняття зафіксованої частини; оскільки символ пропуску є значущим символом тексту, то припинити операцію на цьому кроці можна лише за клавішею Esc; 5) під час уведення та редагування тексту можна змінювати в довільний момент параметри шрифту через FormFontAttr, кольори шрифту (ліва клавіша мишки) і тла (права клавіша мишки) через палітру кольорів внизу вікна, спосіб зафарбування тла через кнопки панелі параметрів зафарбування PanelBrushAttr; кольори і спосіб зафарбування демонструється у правому боці вікна FormFontAttr; 6) зафіксувати текст можна або натисканням мишкою поза межами виокремленої ділянки, або вибором будь-якої іншої операції через кнопки MainToolBar.

Для детального вивчення реалізації операції друкування текстових підписів необхідно уважно розглянути цей процес виконання редактора за вибором команди друкування текстового підпису.



Рис. 4. Форма атрибутів шрифту









### Вибір параметрів текстових підписів

Форма та відповідний модуль призначені для визначення атрибутів текстових підписів: назва шрифту, розмір шрифту, стиль тощо. Форму атрибутів шрифту показано на рис.4. Крім того, у процесі роботи редактора при висвічуванні форми атрибутів шрифту можна визначати також кольори самого шрифту і тла, які обирають з палітри кольорів знизу головної форми. Обрані кольори демонструються з правого боку форми атрибутів.

## Проектування команд графічного редактора

### Вибір зображень курсорів

Зовнішній вигляд курсора допомагає краще сприймати процес малювання, зокрема виконання відповідних операцій. Тому доцільно проектувати перелік курсорів для їхньої реалізації в графічному редакторі, беручи за основу команди редактора та загальноприйняті способи програмування інтерфейсу системних програм. Наприклад, маємо такі курсори:

Малюнок курсора	Кодування в програмі	Функціональне призначення
	crStandard	Малювання стандартних фігур і виконання стандартних операцій
	crCreateTxt	Формування текстових підписів на малюнку
	crAirBrush	Розпилювач – генерація випадкових цяток
	crSelectCol	Вибір кольору безпосередньо за пікселем малюнка (на відміну від вибору через палітру)
	crDrawLine	Малювання довільної лінії за рухом курсора мишки
	crMovePasted	Виконання операцій зовнішньої вставки елементів малюнка (з файлів та з буфера Windows)
	crEraseNull	Фіктивний курсор (без зображення) для операції стирання (“гумка”)
	crNonActive	Графічний редактор не активний

Курсори можна створювати довільним редактором ресурсів Windows, однак зазвичай використовуємо інтегрований редактор Image Editor (якщо такий є). При проектуванні курсорів важливо правильно вибрати кольори з огляду на тло під курсором, яке може бути довільного кольору. Курсор повинен бути видимим на будь-якому тлі.

### Список команд головного меню

На рис.5 показано повний список команд головного меню редактора.

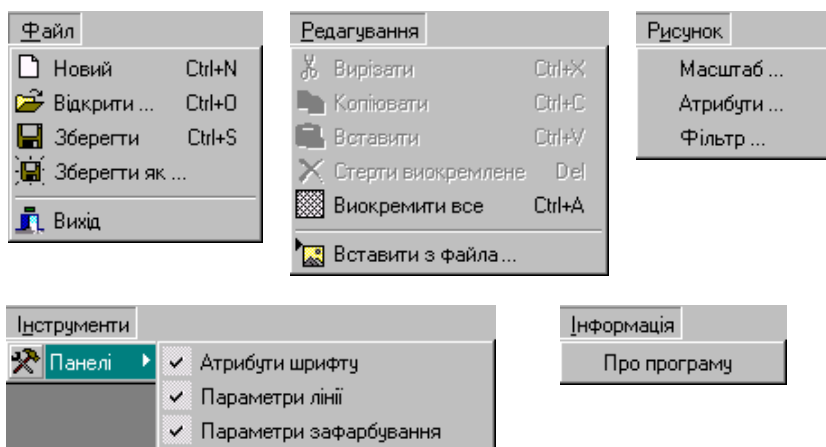


Рис. 5. Список команд головного меню

Крім команд меню операції виконують за допомогою піктографічних кнопок панелей інструментів і параметрів.

### Визначення розмірів малюнка

Форма атрибутів малюнка (рис.6) призначена для визначення ширини та висоти малюнка. У процесі визначення відповідні процедури виконують контроль за правильним записом атрибутів і за дотриманням обмежень щодо розмірів.

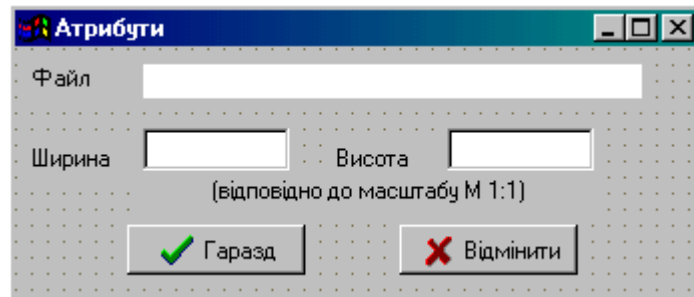


Рис. 6. Форма вибору розмірів малюнка

### Вибір масштабу подання малюнка

У процесі редагування можна подавати малюнок у збільшеному масштабі з метою точного малювання дрібних елементів малюнка. Виглядає доцільним обмежитись фіксованим переліком збільшення масштабу в 2, 4 чи 8 разів. Крім того, масштабування кратне степеням 2 дозволяє зберігати якість растрового формату при зменшенні назад, бо непотрібна апроксимація пікселів. Проект форми для вибору масштабу подано на рис.7.



Рис. 7. Форма до вибору масштабу подання малюнка

Варто зробити важливе застереження. Збільшення масштабу малюнка в  $n$  разів може вимагати збільшення пам'яті комп'ютера в  $n^2$  разів для його зберігання та опрацювання (за растровими форматами) – залежно від способу зберігання. Тому до збільшення масштабу треба підходити виважено, передусім для великих за розмірами малюнків.

### Опрацювання малюнків способом фільтрування

Фільтрування – спеціальний спосіб опрацювання малюнка шляхом перетворення кольорів. Перетворення виконується за певними формулами, що дають змогу отримати різні ефекти. При перетворенні виконується аналіз сусідніх ділянок малюнка (пікселів) довкола кожної точки обраної ділянки малюнка. Крім того, перетворення виконується окремо для кожного з трьох складових кольорів стандартного пікселя: червоного, зеленого, синього. На рис.8 показано форму для вибору деяких типів фільтрів, реалізованих в редакторі.

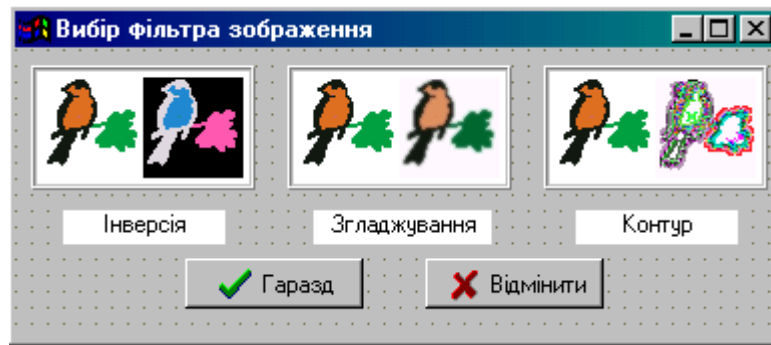


Рис. 8. Форма для вибору типу фільтра