**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ФРАНКА**

**ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ ТА ІНФОРМАТИКИ**

**ЗВІТ**

**З навчальної практики**

**Про виконання проекту**

**"Програмний продукт для роботи з**

**двовимірними многокутниками"**

**Виконали:**

**Студенти 2-го курсу**

**Групи ПМп-22**

**Команда "vVv"**

**Львів-2017**

**ЗМІСТ**

**-----------------------------------------------------**

**Постановка завдання --------------------------------------1 ст.**

**Шляхи реалізації--------------------------------------------1 ст.**

**Хід виконання------------------------------------------------2 ст.**

**Особливості продукту--------------------------------------2 ст.**

**Перелік команд----------------------------------------------3 ст.**

**Інструкція з використання-------------------------------3 ст.**

**Презентація результатів-----------------------------------4 ст.**

**Список учасників, та їх внески--------------------------5 ст.**

**Постановка завдання:**

Створити програмний продукт для роботи з двовимірними многокутниками. Проектування включає опис класів, які необхідні для імплементації необхідної функціональності (очевидно, що, окрім класу "многокутник", як мінімум, необхідно буде ввести клас "точка").

Функціональність:

1. Додавання многокутників: результат - новий многокутник ;
2. Віднімання многокутників: результат - новий многокутник ;
3. Клонування многокутника: результат - новий об'єкт многокутника, що має абсолютно такі ж параметри '
4. Масштабування многокутників: результат - "розтягнутий" або "стиснений" многокутник відносно центру мас;
5. Масштабування многокутників з клонуванням: результат - "розтягнутий" або "стиснений" многокутник відносно центру мас ;
6. Перенесення многокутника: результат - зміщений на певні величини по X та Y многокутник ;
7. Перенесення з клонуванням многокутника: результат - зміщений на певні величини по X та Y многокутник .

**Шляхи реалізації:**

Для створення многокутника нам потрібні точки , та лінії , тому ми створили 3 класи : Polygon, Line, Point , - які є взаємоповязані. Многокутник складається з ліній , а лінії з точок (вершин многокутника). Тому при створенні об'єкта многокутника, ми використовуємо створення таких об'єктів як ліній та точок, які входять у цей многокутник. Фактично, ми реалізували створення многокутника за масивом ліній. Для відображення самого многокутника, ми вирішили використати стандартну бібліотеку "windows.h".

Деякі функції, такі як множення на скаляр (розтягування та стиснення), клонування, зміщення , ми реалізували як перевантаження операторів \*, = ,+=.

Також ми вирішили створити консольне меню, яке дозволяє виконувати необхідні операції, за введенням команд. Список цих команд відображається, при введені "help".

**Хід виконання:**

Спочатку було створені наш основний "кістяк, або скелет" - це самі класи Polygon, Line, Point. Так у класі Point були реалізовані методи необхідні для визначення положення точки (координати X та Y); у класі Line реалізовано вже безпоседеднє використання двох точок для створення лінії (A і B відповідно початок і кінець лінії), також було реалізована функція для її відображення; у класі Polygon було створено масив ліній створеного многокутника, за якими він відображався.

Для завершення даного завдання було створено наступні методи необхідні для повної функціональності програми: move x або y, move point, zoom, draw.

Для зручності використання було створене консольне меню зі застосуванням цих команд (реалізовано у функції devide).

**Особливості продукту:**

Як раніше згадувалось, многокутник будується на основі масиву ліній. Відображається многокутника відбувається на створеній координатній площині відповідно від знайденого центру консолі. Реалізовано консольне використання команд для простішого застосування.

**Перелік команд:**

**"--**p new" - створення многокутника;

"--p move\_x" - зміщення по X;

"--p move\_y" - зміщення по Y;

"--p move\_point" - зміщення по X і Y;

"--p zoom" - масштабування;

"--c draw" - відображення на координатній площині.

**Інструкція з використання:**

Після запуску програми, за допомогою введення команди "help" переглядаємо доступні команди, та як їх використовувати. **ЗАУВАЖЕННЯ:** вводити з точністю до знаків!

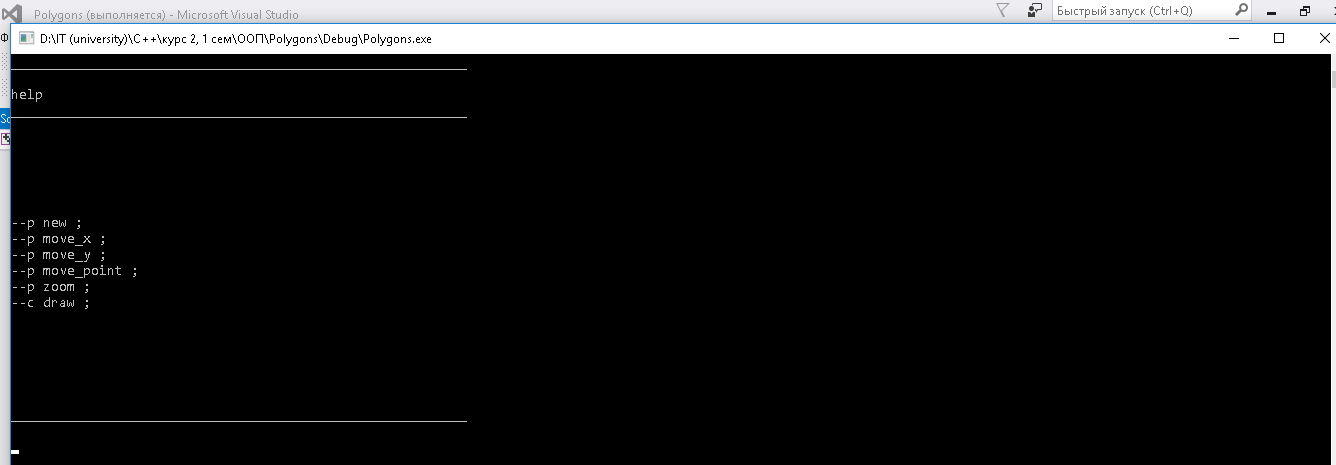
При введенні "--p new" необхідно ввести кількість вершин створеного многокутника, та ввести їх координати відповідно як x1,y1 потім x2,y2 і т.д. до xn,yn, де n - кількість вершин.

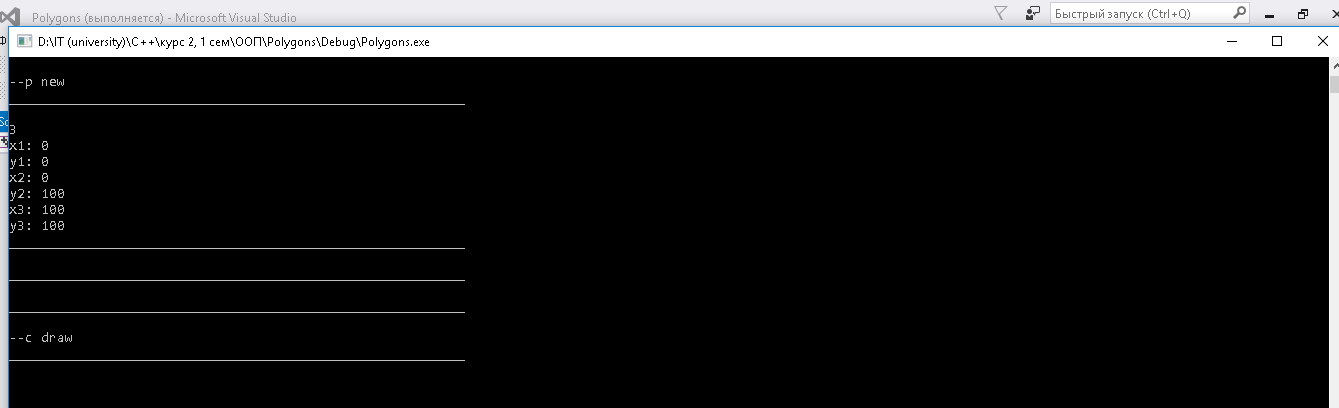
Далі вводимо необхідні команди.

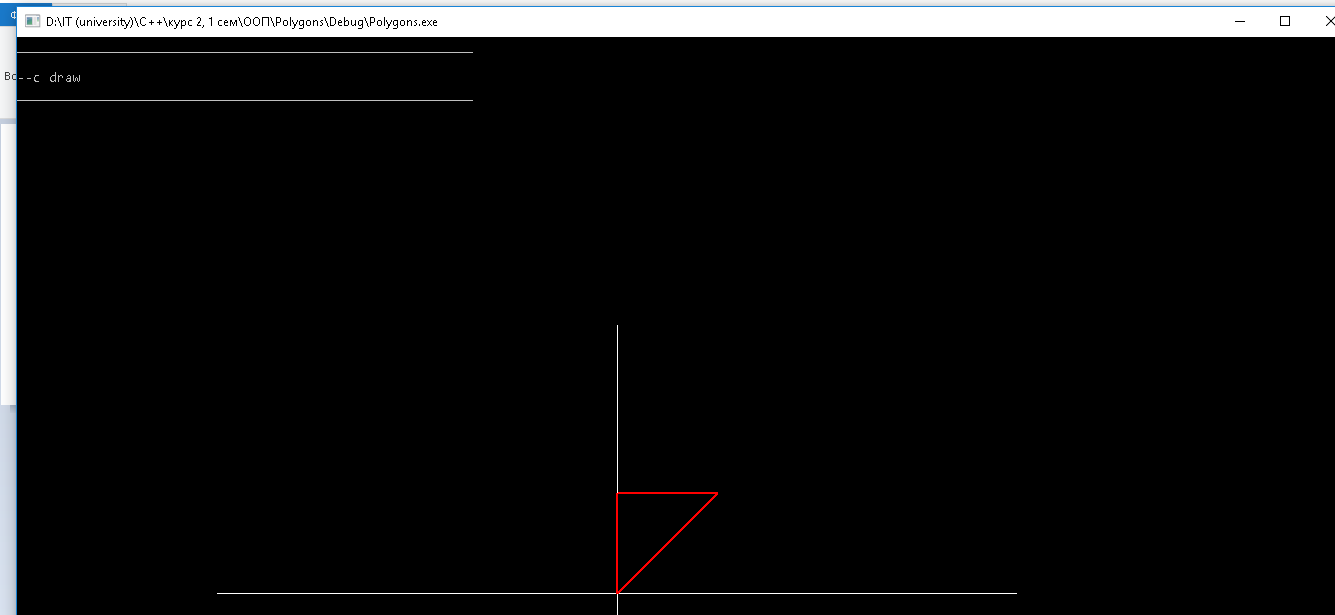
"--p move\_x" та "--p move\_y" вводимо число зміщення.

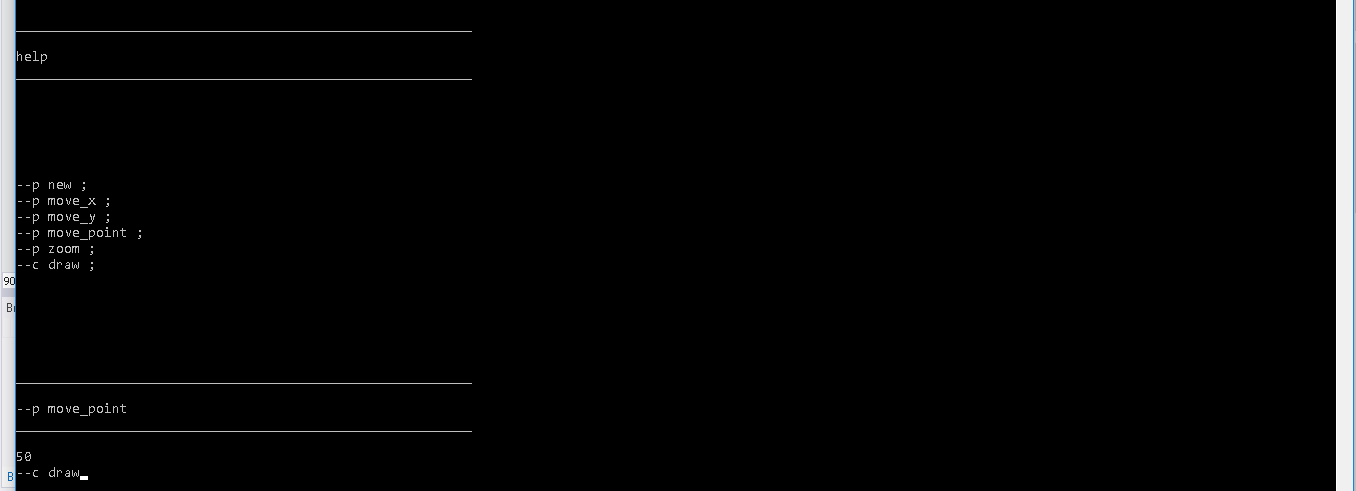
"--p move\_point" вводимо координати зміщення (пара чисел).

**Презентація результатів:**









**Список учасників, та їх внески для реалізації:**

Настосяк Віталій - класи Polygon, Line, Point, функції devide і меню, draw.

Стахів Юлія - operator +, operator += .

Гаєвська Ірина - operator \*, operator \*=.

Лозинський Степан - operator =, меню.