Є.Ф. Вінниченко, А.О. Костюченко
Чернігівський педагогічний університет

Деякі особливості геометричних перетворень в програмі GRAN-2D

Геометричні перетворення – дуже важливий розділ курсу геометрії. У геометрії Евкліда, що вивчається в шкільному курсі математики, переважно досліджуються ті властивості геометричних фігур, які не змінюються при їх русі (образно кажучи, кожну геометричну фігуру можна розглядати як "тверду", наприклад, вирізану з картону), – симетрія та поворот, а також ті, де відбувається перетворення подібності – гомотетія.

Метод геометричних перетворень є досить продуктивним методом розв'язування геометричних задач. Математична теорія симетрії, симетрія у живій та неживій природі, мистецтві, архітектурі, інженерії отримали спільне підґрунтя у геометричних перетвореннях.

Між тим, при навчанні даної теми вчитель зустрічається з певними труднощами у своїй педагогічній практиці. Це може бути пояснено навіть не стільки браком часу, що відводиться на розгляд даної теми, скільки обмеженістю відповідної наочності, що і викликає труднощі у сприйманні матеріалу. Не завжди вчитель має змогу підготувати достатню кількість моделей, що ілюструють відповідний теоретичний або задачний матеріал, тим більше, що як правило такі моделі не відрізняються суттєвою різноманітністю.

Одним із шляхів вирішення цієї проблеми є використання на уроках математики комп'ютера з відповідним програмним забезпеченням. Даний напрямок застосування комп'ютерного моделювання в геометрії широко відомий в нашій країні та за кордоном: достатньо відмітити такі широко відомі програмні продукти як Cabri, Cinderella, DG, GRAN-2D [3,5].

Однак, необхідно відмітити, що наявність відповідного програмного продукту не призводить автоматично до зростання рівня навченості учнів або їх зацікавленості у вивченні предмету. Необхідне врахування й інших чинників: зокрема, відповідність розглядуваного програмного продукту шкільному курсу математики, наявність в цьому програмному продукті зручних інструментів, використання яких дозволяє на якісно новому рівні підходити до вивчення матеріалу, а також наявність методичного забезпечення, орієнтованого на навчання з використання такого продукту.

Природньо, програмний продукт не може бути статичним, він повинен розвиватися: розробники повинні враховувати появу нових та розвиток вже існуючих технологій (мультимедійних, мережевих тощо), нові педагогічні та методичні ідеї. У відповідності до цього в програму GRAN-2D (версія 2.0, 2007 рік) були додані нові засоби опрацюванню геометричних фігур на площині, що дозволить значно ефективніше використовувати дану програму в навчальному процесі. Загалом в даній версії програми відбулося чимало змін та доповнень, порівняно з попередньою. Зокрема вони стосуються інтерфейсу програми, роботи з окремими об'єктами (точками, відрізками, прямими тощо), з'явилися нові об'єкти та послуги. В даній роботі розглядаються деякі нові можливості використання програми GRAN-2D, що стосуються саме вивчення геометричних перетворень на площині, хоч при цьому будуть використовуватися й інші оновлені послуги.

Як вже згадувалось вище, серед значної кількості різноманітних геометричних перетворень, в шкільному курсі математики розглядають лише симетрію (відносно точки та прямої), поворот, паралельне перенесення та гомотетію. Всі ці перетворення геометричних об'єктів можна робити і за допомогою програми GRAN-2D (крім того, в програмі доступні такі геометричні перетворення, як інверсія та деформація, які не розглядаються в шкільному курсі геометрії).

Що стосуються симетрії відносно точки і прямої та паралельному перенесенні, то дані послуги програми практично не змінилися у порівнянні з попередньою версією, а всі нововведення будуть детально розглянуті нижче при розгляді повороту.

Потрібно зазначити, що в програмі передбачається чітке розмежування між симетрією відносно прямої, променя або відрізка. Так, якщо віссю симетрії є пряма, то симетрична точка завжди буде існувати. Якщо ж віссю симетрії є промінь або відрізок, то симетрична точка буде існувати лише тоді, коли перпендикуляр, опущений із заданої точки на пряму, що визначає вісь симетрії (і яка включає в себе вказаний промінь або відрізок), перетинає цей промінь або відрізок. На рис.1 можна бачити, що для точки S_1 існують симетричні точки як відносно прямої AB, так і відносно відрізка CD. Однак для точки S_2 існує лише симетрична точка відносно прямої AB і не існує симетричної відносно відрізка CD.

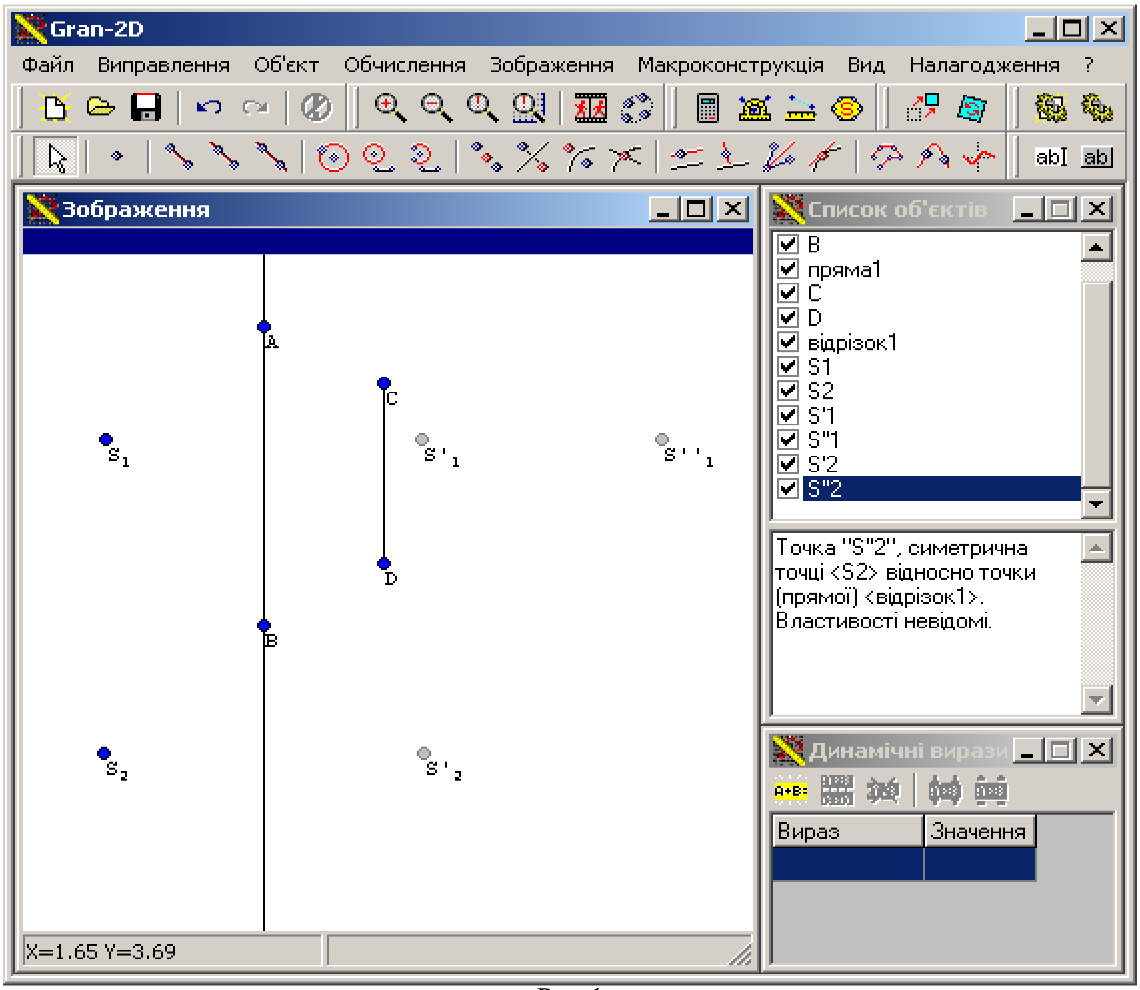

Рис.1

Якщо тема симетрії або паралельного перенесення є відносно простою для розуміння учнями, то при вивченні теми повороту виникають певні складнощі. Саме тому детальніше зупинимось на деяких особливостях побудови комп'ютерних моделей в програмі GRAN-2D, що можуть бути цікавими при розгляді даного питання.

Перетворення однієї фігури в іншу називають ***рухом****, якщо воно зберігає відстань між точками, тобто переводить будь-які дві точки A і B першої фігури у точки A_1, B_1 другої фігури так, що $AB=A_1B_1$.* ***Поворотом площини*** *навколо даної точки називається такий рух, при якому кожний промінь, що виходить з даної точки, повертається на один і той самий кут в одному і тому самому напрямі. Цей кут називається кутом повороту* [4].

Для дослідження властивостей повороту підготуємо зображення, згідно рисунку 2. Спочатку розмістимо дугу, яка в подальшому буде задавати кут повороту (на рисунку такою дугою є дуга, що визначається точками *A*, *B*, *C*). Крім того розмістимо точку *O*, яка буде визначати центр повороту, та примітивні фігури над якими буде виконано поворот: точку (точка *F* на рисунку), відрізок, промінь, пряму, коло, замкнену ламану.

Спочатку розглянемо найбільш простий випадок: поворот точки навколо іншої точки. Для цього створимо точку, яка буде результатом повороту точки *F* навколо точки *O* на кут *ABC*. Це можна зробити, створивши аналітичну точку (команда ***"Об'єкт / Створення / Аналітична точка"***). Координати створюваної точки задамо наступними виразами:

$$X = (X(F)-X(O))*Cos(OAngle(A,B,C))-(Y(F)-Y(O))*Sin(OAngle(A,B,C))+X(O),$$
$$Y = (X(F)-X(O))*Sin(OAngle(A,B,C))+(Y(F)-Y(O))*Cos(OAngle(A,B,C))+Y(O).$$

Ці формули відповідають формулам повороту довільної точки з координатами $(x;\ y)$ навколо заданої точки з координатами $(x_0;\ y_0)$ на кут φ:

$$x' = (x-x_0)\cos\varphi-(y-y_0)\sin\varphi+x_0,$$
$$y' = (x-x_0)\sin\varphi+(y-y_0)\cos\varphi+y_0.$$

В результаті даної операції буде створено точку *V*. Тепер, змінюючи кут повороту чи положення опорних точок повороту *O* та *F*, можна бачити що положення результуючої точки *V* відповідним чином змінюється, проте залежність її від опорних об'єктів залишається сталою.

Для виконання повороту більш складного об'єкту необхідно виконати поворот базових точок, що визначають даний об'єкт: для відрізка або прямої – це дві точки, для *n*-кутника – кількість точок буде дорівнювати *n*. Тому використовувати описаний вище спосіб для кожної з точок досить незручно. Існує два шляхи спрощення процесу розв'язування такої задачі.

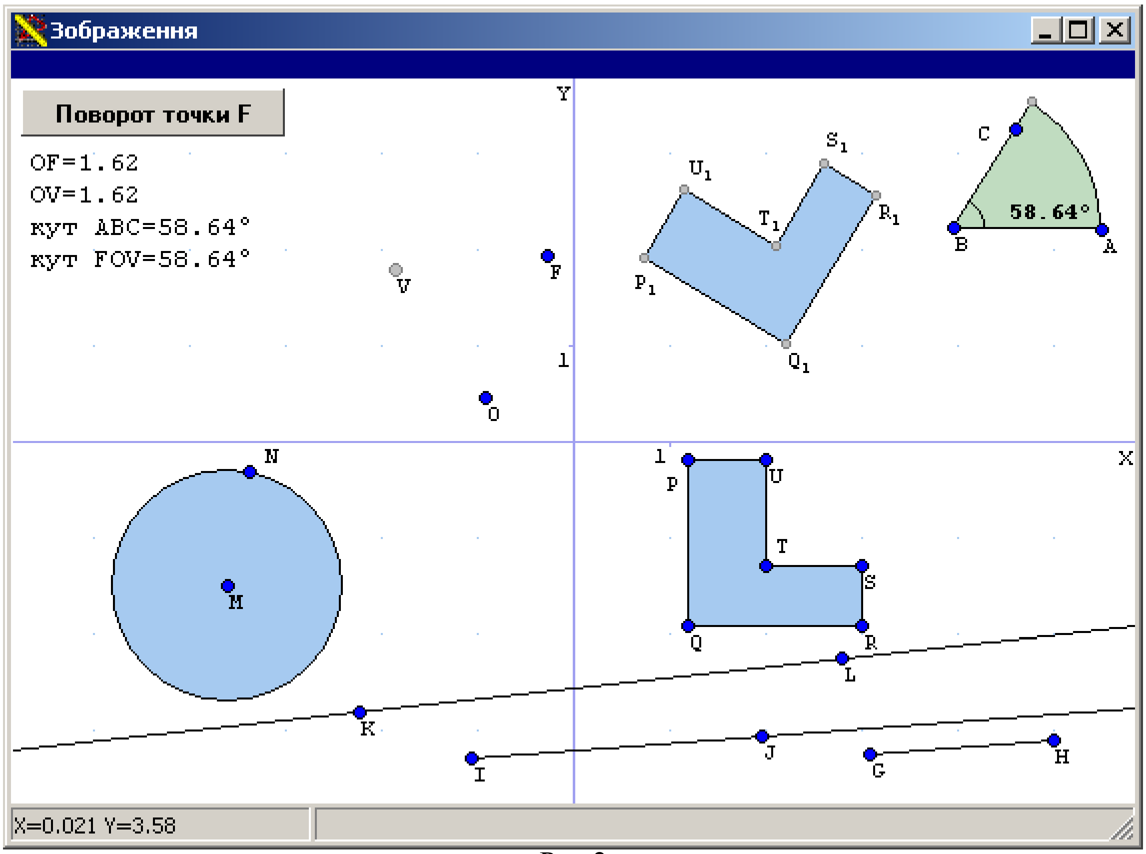

Рис.2

Щоб відобразити результат повороту навколо точки *O* на кут *ABC* інших об'єктів (крім ламаної) можна поступити наступним чином:

- прикріпити точку *F* до відповідного об'єкту (для цього необхідно підвести точку *F* до цього об'єкту, в контекстному меню точки вибрати ***"Прикріпити точку"*** та вказати відповідний об'єкт)
- побудувати геометричне місце точок (скориставшись послугою ***"Зображення / ГМТ"***), вказавши точку *F*, як точку на об'єкті, а точку *V*, як залежну.

Для виконання повороту ламаної можна побудувати відповідні аналітичні точки до всіх опорних точок ламаної, а потім їх з'єднати новою ламаною.

Проте такий спосіб хоча і є дієвим, але виявляється достатньо громіздким, особливо у випадку застосування геометричних перетворень до многокутників. Тому в програмі передбачена й інша можливість для виконання повороту за рахунок автоматизації попередніх дій. Зокрема, за даним методом можна швидко повернути не тільки відрізок, промінь, пряму або коло, а й ламану будь-якої складності.

Для цього слід скористатись послугою ***"Об'єкт / Перетворення параметрично"*** закладинка ***"Поворот"*** (рис.3). Далі необхідно вказати:

- до якого об'єкту буде застосовуватися перетворення;
- точку, що визначає центр повороту;
- вказати кут повороту. Кут повороту вказується в градусах явно або за допомогою формули (рис.3). Додатнім кутом повороту вважається поворот проти годинникової стрілки.

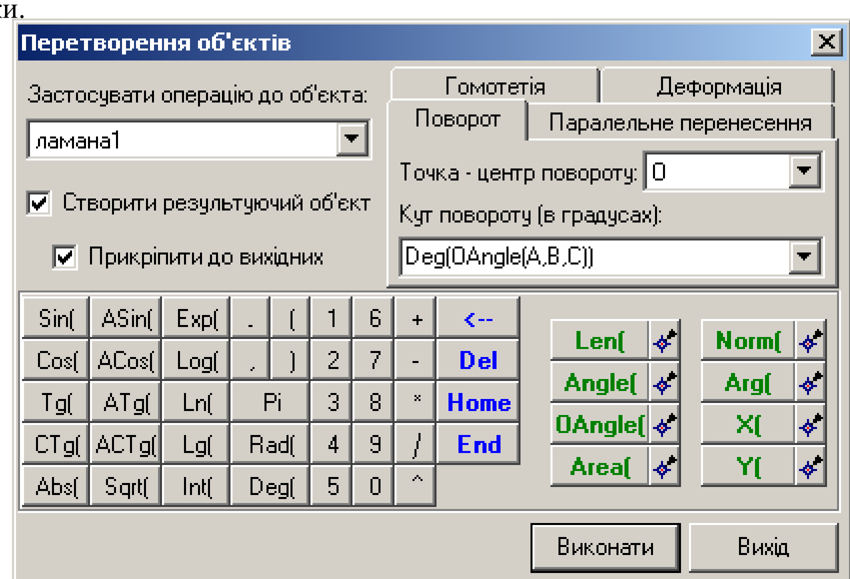

Рис. 3

Оскільки необхідно, щоб в результаті перетворення був створений новий об'єкт і він був зв'язаним (підтримувалася залежність результуючого об'єкту від вихідного), необхідно відмітити послуги: ***"Створити результуючий об'єкт"*** та ***"Прикріпити до вихідних"***.

Як відомо, композиція двох поворотів відносно одного центру є також поворот. Однак, якщо повороти виконуються навколо різних центрів, то відповідь про результат композиції двох таких перетворень не є очевидною. За допомогою комп'ютерних експериментів можна не тільки переконатися, що результатом таких перетворень також є поворот, але й визначати параметри цього повороту. Покажемо це на наступній комп'ютерній моделі.

Для даного дослідження підготуємо зображення згідно рисунку 4. Визначимо кути поворотів *IHL* та *DCG*, відповідні центри поворотів O_1 та O_2, і розмістимо дві вільні точки *A* та *B*. Виконаємо побудови для експериментальної перевірки. Повернемо точку *A* навколо точки O_1 на кут *IHL*, а потім одержану в результаті цього повороту точку A_1 повернемо навколо точки O_2 на кут *DCG*. Результатом є точка A_2. Аналогічні операції проводимо для точки *B* (рис. 4).

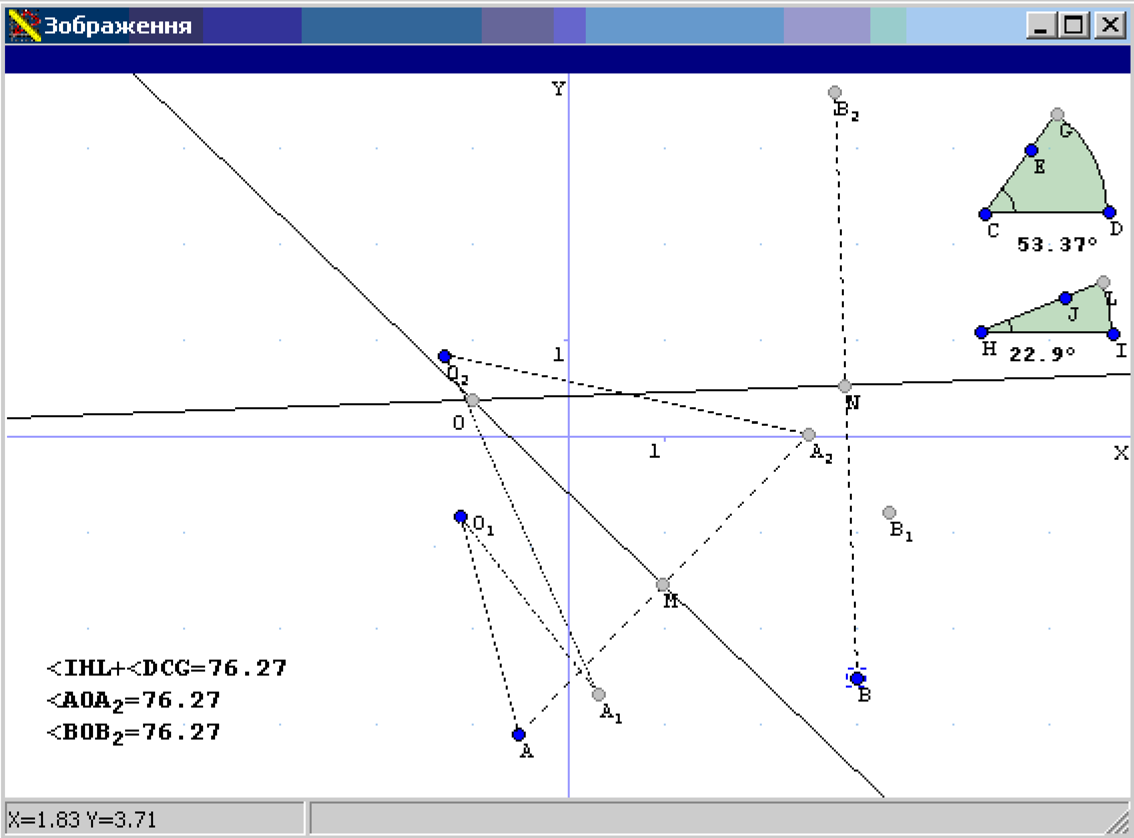

Рис. 4

Припустимо, що точки A_2 та B_2 є результатом повороту точок *A* і *B* навколо деякого центру *O*. Центр повороту можна знайти як точку перетину двох серединних перпендикулярів до пар точок "другий прообраз" – "образ" (тобто до відрізків AA_2 та BB_2). Результати досліджень та вимірів показують, що точки A_2 та B_2 можуть бути одержані з точок *A* та *B* результатом повороту навколо точки *O* на один і той самий кут.

Більш того, можна з'ясувати, що чисельно величина цього кута повороту дорівнює алгебраїчній сумі двох кутів повороту, що утворюють композицію (тобто $\angle AOA_2 = \angle BOB_2 = \angle DCG + \angle IHL$).

Зрозуміло, що перевірка величини кута відбувається з певною точністю [1], проте вірогідність правильності цього факту в цілому дуже велика (можна сказати, що після комп'ютерного підтвердження гіпотези з'являється зовсім нове почуття впевненості у правильності відповідної теореми, особливо в тих учнів, у яких наочно-образний тип мислення превалює над абстрактно-логічним).

Для теоретичного обґрунтування залишається розглянути задачу, що підтвердить експериментальні побудови. Для цього необхідно дослідити конфігурацію, яку утворюють два центри вихідних поворотів O_1 і O_2 та центр гіпотетичного повороту *O*, та скористатися твердженням, згідно якого композиція двох симетрій відносно прямих, що перетинаються, є поворот відносно точки перетину на подвійний кут, що утворюють ці прямі (кут утворюється осями симетрії у напряму проти годинникової стрілки).

Зазначимо, що особливості побудови розглядуваних моделей (наприклад, створення результуючих об'єктів та їх безпосереднє перетворення в залежності від вихідних) є такими ж для симетрії, паралельного перенесення та гомотетії.

Цікавою проблемою для учнів може стати пошук таких фігур, і особливо многокутників, які при обертанні навколо певної точки, переходять самі в себе. Зокрема, серед опуклих многокутників такими є правильні многокутники. Саме правильні многокутники та задачі з ними доволі часто розглядаються в шкільному курсі геометрії. Отже саме моделі таких многокутників доводиться будувати у відповідних програмах.

Якщо побудова правильного трикутника або квадрата є відносно простою, то вже побудова правильних многокутників із більшою кількістю сторін часто викликає труднощі. Розглянемо, зокрема, побудову моделі правильного многокутника на прикладі побудови правильного шестикутника.

1. Будуємо довільний відрізок *AB* - радіус майбутнього кола (рис.5).

2. Будуємо коло за заданим радіусом із центром в довільній точці O, скориставшись інструментом ***"Коло за радіусом"***.

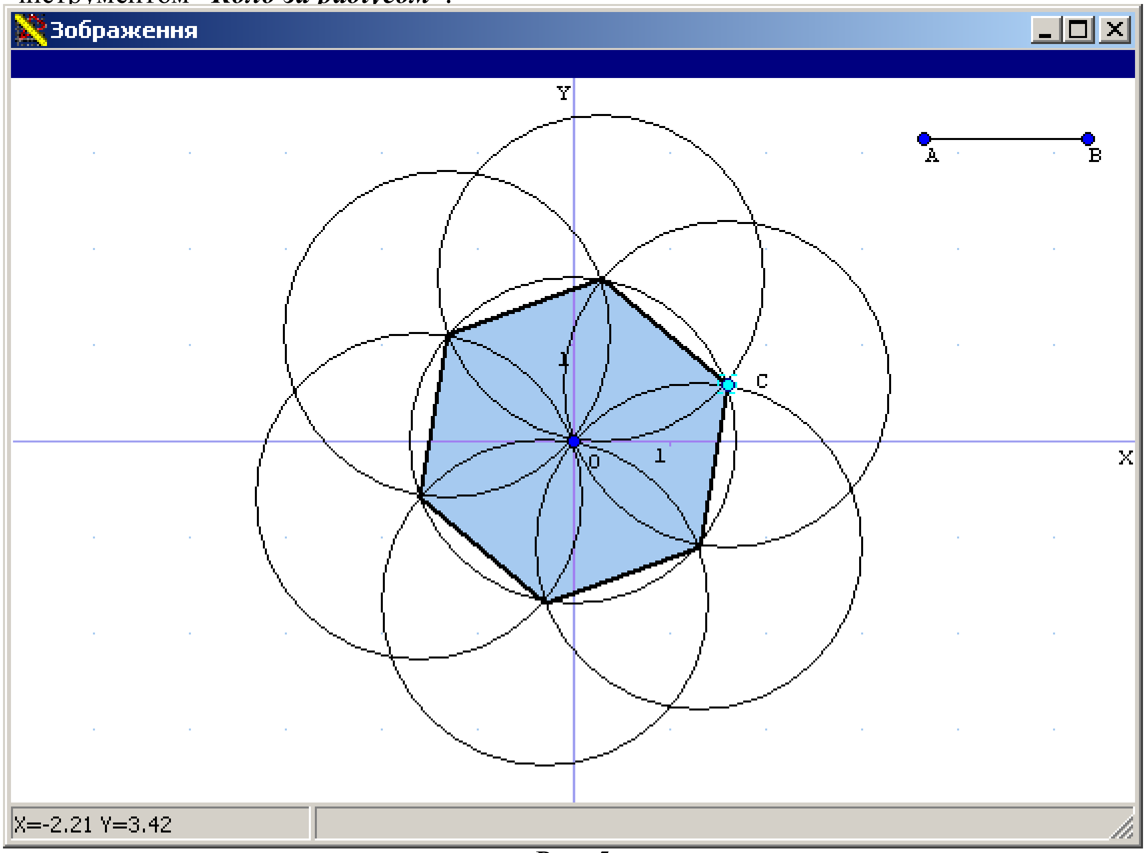

Рис. 5

3. Створюємо точку на колі C, скориставшись інструментом ***"Створення точки"***. Дана точка буде прив'язаною до кола і може вільно рухатись вздовж нього.
4. Відкладаємо від побудованої точки хорду кола, довжина якої дорівнює радіусу кола. Для цього необхідно:
   - побудувати коло за заданим радіусом AB з центром в точці C, скориставшись інструментом ***"Коло за радіусом"***;
   - побудувати точку перетину вихідного кола з побудованим колом, скориставшись інструментом ***"Точка перетину"***.
5. Повторюємо попередній пункт ще 5 разів, приймаючи за вихідну точку кола, яку було побудовано на попередньому кроці.
6. Будуємо многокутник з вершинами в побудованих точках, скориставшись інструментом ***"Ламана"***.
7. Ховаємо всі допоміжні побудови.

В результаті отримаємо зображення правильного шестикутника. Дана побудова відповідає побудові многокутника за допомогою циркуля і лінійки. Крім того, що вона є достатньо громіздкою, такі побудови можуть використовуватися лише для правильних многокутників із числом сторін: 3, 4, 5, 6, 8, 10, 12, 15, 16, тощо. Однак їх не можна використати для побудови правильних многокутників із числом сторін: 7, 9, 11, 13, 14, 18, та інших [5].

Саме тому за основу побудови правильних многокутників можна взяти деякі їх властивості, що стосується кутів таких многокутників. Як відомо, кут між сторонами правильного n-кутника в градусах дорівнює $\frac{180\cdot(n+2)}{n}$, а центральний кут – $\frac{360}{n}$ градусів. Саме ці властивості і використовуються при побудові правильних многокутників в програмі GRAN-2D.

Використовуючи послугу ***"Об'єкт / Створення / Правильний многокутник"***, можна побудувати правильний многокутник, вказавши в допоміжному вікні опорні об'єкти, відносно яких його буде створено: коло (вписане або описане), центр многокутника або його сторона. Тип побудови обирається через закладинку у відповідному вікні (рис. 6).

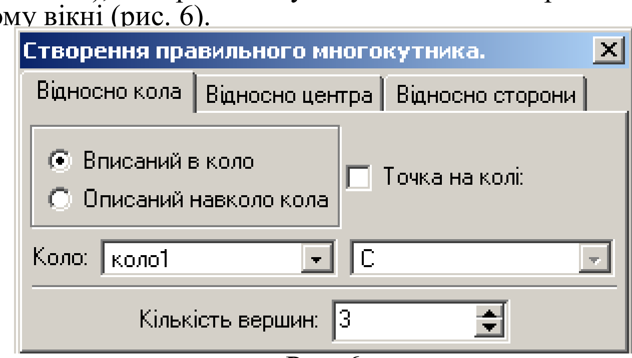

Рис. 6

Для побудови многокутника *відносно кола* необхідно:

- Вказати, вписаним в коло чи описаним навколо кола буде створюваний многокутник;
- вибрати коло, відносно якого буде відбуватися побудова;
- вказати точку на колі, яка буде вважатися початковою (якщо на колі не міститься жодної точки, її буде створено автоматично). Для многокутника, вписаного в коло, дана точка буде однією з вершин. Для многокутника, описаного навколо кола, дана точка буде точкою перетину кола з прямою, що сполучає центр кола і одну з вершин многокутника.

Для побудови многокутника *відносно центра* необхідно вибрати точку, яка буде визначати центр многокутника, та точку, що визначатиме початкову (опорну) вершину. Всі інші вершини многокутника будуть розташовані проти годинникової стрілки відносно опорної вершини.

Для побудови многокутника *відносно сторони* необхідно вибрати точки, що будуть визначати одну з сторін многокутника. Всі інші вершини будуть розташовані за годинниковою стрілкою відносно опорних точок (тут враховується порядок, в якому вказуються кінці відрізка, що визначають сторону многокутника).

Після встановлення всіх опорних точок та параметрів створюваного об'єкта за програмою буде створено кілька нових точок – вершин правильного многокутника, які є результатом повороту опорних точок на визначений кут. Ці точки з'єднуються ламаною, що і визначає правильний многокутник. Потрібно відзначити, що всі залежні вершини многокутника утворюються з опорних шляхом повороту, в чому можна переконатися, проаналізувавши положення будь-якої з цих точок. Зауважимо, що максимальна кількість вершин правильного многокутника, що можна будувати за допомогою цієї послуги, дорівнює 20.

Зручність вказаного способу побудови правильних многокутників спрощує не тільки роботу з програмою, але й побудову відповідних комп'ютерних моделей, дозволяє розглядати більш складні задачі. Наведемо деякі з них (з окремими задачами можна ознайомитись в [2]).

1. Чи може вписаний в коло многокутник мати рівні кути, але нерівні сторони? Рівні сторони, але нерівні кути?
2. Чи може описаний навколо кола многокутник мати рівні кути, але нерівні сторони? Рівні сторони, але нерівні кути?
3. В коло радіуса R вписаний правильний n-кутник. Знайти радіус кола, вписаного в цей n-кутник.
4. В правильному n-кутнику (n>4) провели всі діагоналі, що сполучають вершини через одну. Довести, що точки перетину діагоналей також є вершинами правильного n-кутника.
5. З яких правильних многокутників можна скласти паркет?
6. Чи можна скласти паркет з правильних восьмикутників та квадратів? Правильних 12-кутників та трикутників? Правильних десятикутників та п'ятикутників?

Загалом, останні дві задачі є прикладами відомого класу задач на складання паркетів та мозаїк. Їх розв'язуванням як для правильних, так і для довільних опуклих та неопуклих многокутників багато століть займалась значна кількість вчених-математиків. До речі, деякі підкласи задачі пошуку багатокутників, з яких можна складати паркети та мозаїки, досі лишаються нерозв'язаними. Безперечно, такі задачі, в поєднанні з потужним інструментом для експериментальних досліджень, яким є комп'ютерні моделюючі програми, стануть цікавими і для сучасних учнів.

ЛІТЕРАТУРА

1. Вінниченко Є.Ф. Деякі особливості використання математичних програмних засобів на уроках математики // Комп'ютерно-орієнтовані системи навчання. Зб. наук. праць / Редкол. – К.: НПУ ім.М.П.Драгоманова. – Випуск 6. – 2003. – С. 152-161.
2. Делоне Б., Житомирский О. Задачник по геометрии. – М.: ОГИЗ, Гостехиздат, 1949. – 304 с.
3. Жалдак М.І., Вітюк О.В. Комп'ютер на уроках геометрії. – К.: РННЦ „ДІНІТ", 2004. – 154 с.
4. Погорєлов О.В. Геометрія: Підручник для 7-11 класів середньої школи. – К.: Радянська школа, 1992. – 352 с.
5. Раков С.А, Горох В.П, Осенков К.О., Думчикова О.В., Костіна О.В., Ларін О.Р., Лисиця В.Т., Олійник Т.О., Пікалова В.В.. Відкриття геометрії через комп'ютерні експерименти в пакеті DG. – Харків: ХДПУ, 2000. – 202 с.