

### **Сценарний підхід в навчанні інформатики: об'єктність, наочність, креативність**

**Цілі та рівні шкільного курсу інформатики.** Курс інформатики був введений в шкільну програму СРСР у 1985 році в якості обов'язкової дисципліни в рамках реформи інформатизації освіти, яку очолив академік А.П. Єршов. Обґрунтовуючи необхідність такої навчальної дисципліни в школах, А.П. Єршов вважав, що в модель випускника загальноосвітньої школи, який вступає в життя в епоху інформаційного суспільства, необхідно закласти вимоги до знань, умінь і навичок виходячи з досвіду діяльності програміста [1]. Тоді використовувався термін «програмістський стиль мислення», а зміст шкільного курсу інформатики базувався на метафорі, сформульованій А.П. Єршовим в 1981 році «Програмування – друга грамотність» [2]. Пізніше А.П. Єршов запропонував замінити «програмістський стиль мислення» на «операційний», підкреслюючи, що складовими характеристиками операційного стилю мислення є конструктивність (алгоритмічна складова) та структурованість (вміння підрозділяти складні дії на складові його операції).

У подальшому на пострадянському просторі зміст шкільного курсу інформатики змінювався в залежності від вимог часу. Як зазначає академік М.І. Жалдак в [3], в 1990-х роках відбувся перехід від програмістського підходу в навчанні до користувацького, оскільки стало зрозуміло, що необхідно навчити учнів користуватися готовими програмними продуктами. Були запропоновані різні програми курсу інформатики загальноосвітніх шкіл [4]:

1. Універсального профілю (10-11 класи) [4, 3-15];
2. Без використання комп'ютерів (10-11 класи) [4, 16-28];
3. Філологічного та суспільно-гуманітарного профілів (10-11 класи) [4, 29-40];
4. Художньо-естетичного профілю (10-11 класи) [4, 41-52];
5. Спортивного профілю (10-11 класи) [4, 53-64];
6. Фізико-математичного, природничого та технологічного профілів (10-11 класи) [4, 65-84];
7. Технологічного профілю (7-11 класи) – дисципліна називалася «Інформатика і технології» [4, 85-139];
8. Спеціалізованих шкіл, гімназій, ліцеїв (поглиблений курс; 8-11 класи) [4, 140-183].

В даний час зі шкільного курсу інформатики в Україні введено нові навчальні програми, представлені на сайті МОН [5], [6], [7]. За основу береться підхід, запропонований І.О. Завадським, Ю.О. Дорошенком, Ж.В. Потаповою [5]. Рекомендується обов'язкове навчання інформатики у загальноосвітніх школах у 9-12 класах. У 10-12 класах програма поділена на два рівні:

1. Рівень стандарту [6] – для загальноосвітніх шкіл усіх профілів суспільно-гуманітарного, філологічного, художньо-естетичного та спортивного напрямів, а також для фізичного, біолого-хімічного, фізико-хімічного й агрономічного профілів природничо-математичного напряму обсягом 1 година на тиждень.
2. Академічний рівень [7] – для загальноосвітніх шкіл математичного, фізичного та фізико-математичного профілів природничо-математичного напряму в обсязі 1 година на тиждень у 10-11 класах та 2 години на тиждень у 12 класах.

У нових навчальних програмах в якості мети навчання курсу інформатики відзначено «формування у учнів теоретичної бази знань з основ інформатики, умінь і навичок ефективного використання сучасних комп'ютерно-інформаційних технологій у своїй діяльності», а також формування у випускників школи основ «інформаційної культури та інформатично-комунікативної компетентності». Крім того, у нових навчальних програмах більший акцент робиться на освоєння програмування.

**Необхідність нових підходів.** Наприкінці 1990-х років, а також на початку нового тисячоліття комп'ютерні спеціальності були одними з найпопулярніших серед абітурієнтів. Проте останнім часом намічається тенденція зменшення кількості абітурієнтів, що бажають поступати на комп'ютерні спеціальності. Крім того, рівень знань студентів-першокурсників, що надійшли на ці спеціальності, нерідко недостатній для оволодіння азами програмування при методиках навчання, які використовувалися раніше. У зв'язку з цим у процесі підготовки студентів першого курсу необхідно використовувати нові підходи до навчання програмування, а далі, після апробації і налагодження цих підходів до початкових університетських курсів інформатики, вибудовувати модифікацію та адаптацію їх вниз за віковими сходами, створюючи нові перспективи для шкільної інформатики.

Існують різні педагогічні підходи, які використовуються при навчанні програмування новачків. Найбільш відомі підходи (їх шість) [8]: імперативний, об'єктний, функціональний, широкий, алгоритмічний і апаратний. В даний час широко застосовується об'єктний підхід, який передбачає використання принципів об'єктно-орієнтованого програмування та проектування з перших днів навчання студентів програмування.

**Сценарний підхід Alice.** Зазвичай при навчанні програмування початківцям показують ілюстрації алгоритмів і мовних конструкцій на прикладах з різних предметних галузей, наприклад,

бізнесу, економіки і тому подібне Ми застосовуємо інший підхід – студентам першого року навчання пропонується вивчати основи програмування, розробляючи анімаційні фільми і прості відеоігри у спеціально створеному для цих цілей середовищі програмування Alice [9], [10], [11]. Студенти управляють поведінкою тривимірних об'єктів і персонажів у віртуальному світі. Програмний продукт Alice є продуктом open source (з відкритим програмним кодом); всі його версії безкоштовні і доступні на сайті Alice.org [10]. Тут описано застосування цього підходу – підходу, надихаючого учнів своєю креативністю, нечутливістю до помилок новачка, підходу, ефективно понижуючого бар'єри у вивченні програмування – в навчанні студентів першого курсу. У нас немає ні найменших сумнівів у тому, що творча провокативність цього підходу дає змогу легко перенести його (після необхідної «доместикації») у навчання шкільної інформатики.

Основні положення Alice:

1. Дерево об'єктів (відмічено цифрою 1 на рис. 1) відображається у вигляді списку об'єктів в поточному світі Alice і дозволяє студентам вибирати об'єкти.
2. Редактор сцен (відмічено цифрою 2 на рис. 1) дозволяє студентам розміщувати об'єкти в їх 3D-світі.
3. Студенти використовують події для того, щоб зв'язати метод з дією миші, колізією об'єкта і т.п. Це поле відзначено цифрою 3 на рис. 1.
4. Область деталей (відмічено цифрою 4 на рис. 1) відображає методи, функції та дані для обраних об'єктів.
5. Студенти можуть створювати програми перенесенням методів з області деталей. Поле методів зазначено цифрою 5 на рис. 1.

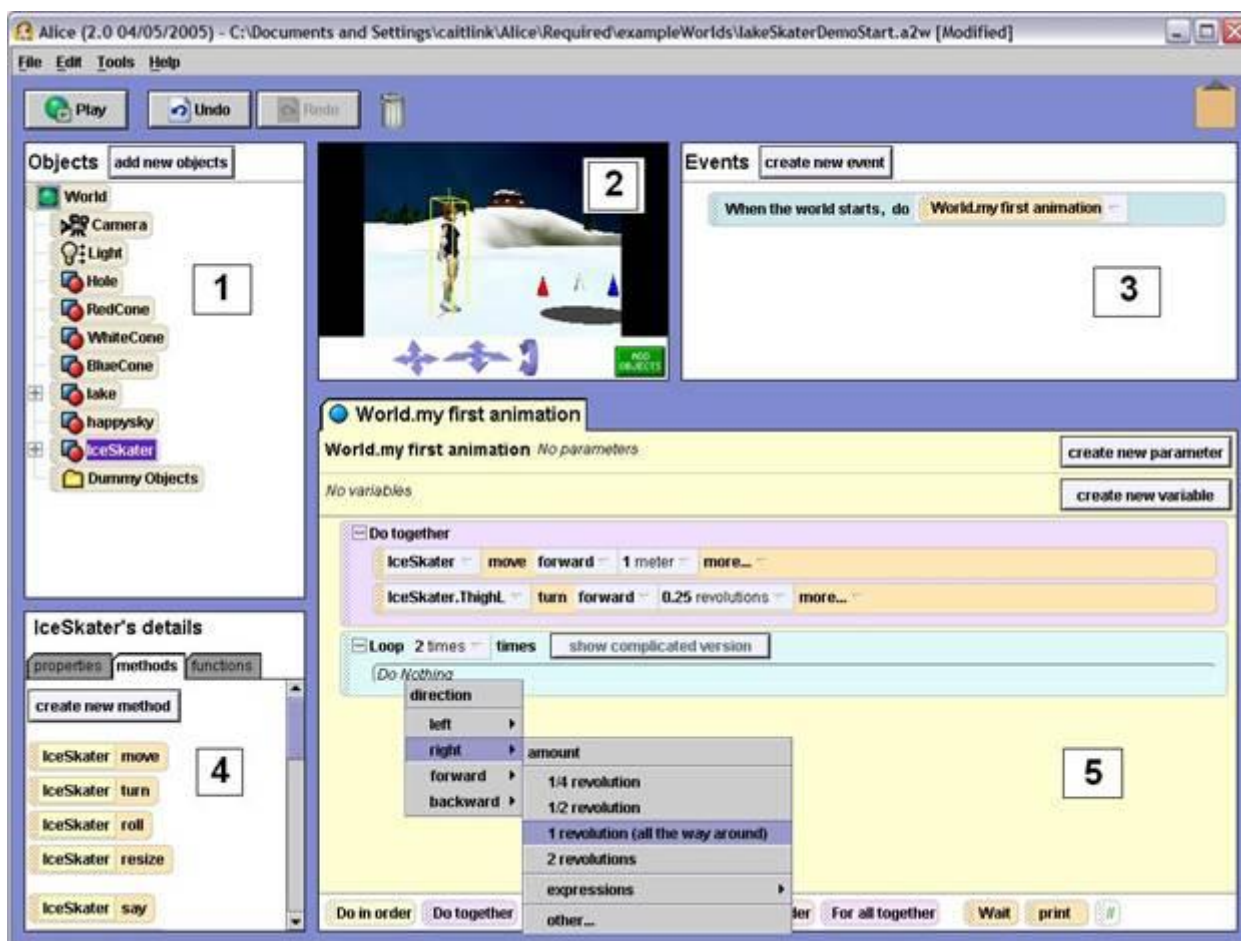


Рис 1. Головне вікно Alice

У Alice студенти будують програми, використовуючи технологію «dragging and dropping», тобто «перетягуванням» іконки, плитки, які відповідають словам в мові програмування. У Alice студенти не можуть допускати синтаксичних помилок (що є звичайно джерелом розчарування студентів, які тільки починають освоювати ази програмування) – вони автоматично усуваються за програмою. Студенти можуть бачити, як створена ними програма виконується; бачити помилки, які вони допустили. На відміну від інших середовищ програмування для початківців, в Alice студенти можуть отримати досвід роботи з усіма конструкціями програмування, які зазвичай вивчаються в курсах програмування.

Alice вважається вільно поширюваною об'єктно-орієнтованою мовою програмування з відкритим кодом і з інтегрованим середовищем розробки IDE (an Integrated Development Environment), яка використовується для навчання початківців програмістів. Вона реалізована на мові програмування Java. Програмний продукт Alice розробляється дослідниками університету Карнегі

Меллон (Carnegie Mellon University). Alice була розроблена для розв'язання трьох основних завдань у навчанні програмування студентів (особливо тих, у кого слабка математична підготовка і/або відсутній досвід програмування):

- Більшість промислових мов програмування є досить складними для студентів першого року навчання. *Alice призначена для навчання програмування новачків.*
- Alice об'єднана з IDE, тому студентам *немає необхідності запам'ятовувати синтаксис.* Також в Alice повністю підтримується об'єктно-орієнтоване програмування, подієво-орієнтоване програмування.
- *Alice розрахована на школярів, студентів першого року навчання.*

Педагогічним інструментарієм у сценарному підході є у даному випадку використання інноваційного підходу при підготовці студентів першокурсників комп'ютерних спеціальностей. Головну складову курсу складає програмне забезпечення «Alice», де використовується 3D-моделювання в дружньому для новачка інтерфейсі, з подальшим введенням основних понять об'єктно-орієнтованого програмування.

Метою навчання дисципліни «програмування для початківців» є набуття знань та практичних навичок використання комп'ютера та програмного забезпечення для створення та опрацювання графічних зображень, використання комп'ютера при 3D-моделюванні, на прикладі вивчення програмного продукту «Alice».

Дане середовище програмування можна з успіхом використовувати для реалізації моделей фізичних процесів і створення повчальних, контролюючих, демонстраційних мультиплікаційних роликів. Даний курс виступає як засіб формування наукового світогляду студентів, реалізації цілей і задач загальної освіти галузі комп'ютерних наук.



**Рис. 2.** Вікно перегляду фільму, створеного студенткою першого курсу РВНЗ «КИПУ» Лемарою Меметовою в 2009–2010 навчальному році

Ця початкова стадія навчання програмування є спеціальною технологією реалізації складних систем в робочі програмні продукти. Програма представлена у вигляді об'єктів, оскільки для студентів видніше, як виконується програма, вона легше створюється, головне – із збільшенням складності кінцевої мети залишається добре структурованою, надійною, гнучкою і керованою. Програмування з використанням 3D-моделювання є відмінним введенням в захоплюючий світ створення програм на молодших курсах. Це дуже схоже на використання комп'ютерних ігор при навчанні комп'ютерних дисциплін за допомогою яких відбувається ілюстрація безлічі складних понять інформатики. Але слід зазначити, що на заняттях з дисципліни «Програмування для початківців» здійснюється розповідь історій, що мотивує подальше навчання студентів (рис.2).

**Можливості використання Alice.** В «Alice» пропонується величезна кількість вдосконалених послуг, які задовольняють зростаючим потребам студентів комп'ютерних спеціальностей. Її відкрита архітектура дозволяє студентам скористатися перевагами використання бібліотеки об'єктів, легко

додавати вражаючі ефекти. Більш того, за допомогою «Alice» вони можуть створювати свої власні програмні коди, щоб надавати оригінальний вигляд своїм творінням. Ось лише деякі можливості, використання програми «Alice» при створенні окремих зображень:

- використання і моделювання геометричної форми будь-яких об'єктів;
- реалізація різних способів управління переміщенням або зміною властивостей об'єктів в процесі анімації, забезпечуючи можливість вірогідності імітації різних типів рухів;
- створення зв'язаних ієрархічних ланцюжків об'єктів і їх анімація за методами прямої або зворотної кінематики, коли рух одного об'єкта викликає узгоджені переміщення решти об'єктів ланцюжка.

Вивчення студентами даного курсу треба починати з основних понять, щоб сформувалося уявлення про технологію проектування і розробки сучасних програмних продуктів. У даному курсі студент знайомиться з 3D-моделями об'єктів (наприклад, тварини і транспорт), які заселяють віртуальний світ, забезпечуючи об'єктно-орієнтований «світ». Написавши прості сценарії, студенти можуть управляти появою і поведінкою віртуальних об'єктів [9].

При конструюванні власних віртуальних світів, у студентів обов'язково сформується деяке розуміння 3D-координатної системи і просторового взаємовідношення об'єктів. Курс дозволяє візуально втілювати думку щодо стану робочого середовища студента. Перевага - це те, що, студенти можуть легко бачити поточний стан використання їхньої програми:

- розташування кожного об'єкту;
- колір об'єкту;
- зв'язки з іншими об'єктами.

Немає ніякої необхідності малювати абстрактні картини для вирішення конкретної проблеми.

**Робота в середовищі Alice.** Щоб почати роботу з «Alice» та з її допомогою створити будь-який мультиплікаційний ролик (фільм), необхідно дотримуватися наступних етапів роботи з даним програмним продуктом:

1. Розробка сценарію є найпершим етапом роботи і завершується створенням опису майбутнього анімаційного ролику у вільній бажано літературній формі. Сценарій майбутнього фільму розробляється студентами самостійно згідно з власним бажанням, виходячи з його орієнтовної тривалості 30-40 секунд.

2. Проектування ролика. На цьому етапі необхідно продумати склад сцени. Слід передбачити всі об'єкти і їх деталі, які будуть видимі з передбачуваних напрямів спостереження. При цьому корисно буває намалювати один або кілька ескізів майбутньої сцени (рис. 3).



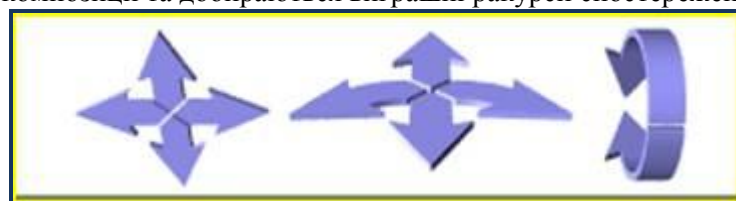
**Рис. 3.** Приклад ескізу майбутньої сцени

3. Формування геометричної моделі сцени. Здійснюється вибір моделі сцени та її об'єктів (рис. 4).



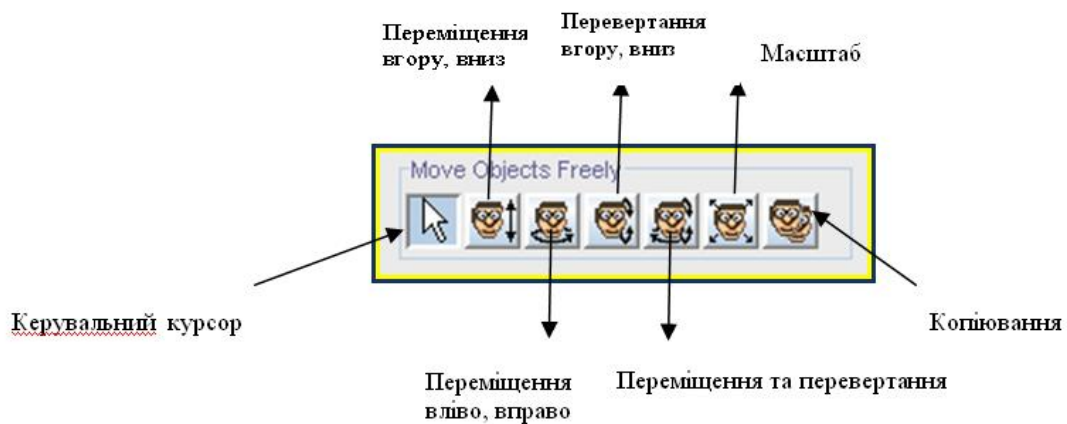
**Рис. 4.** Вибір сцени та об'єктів

4. Налаштування знімальних камер. Відбувається відтворення умов освітлення сцени, що відповідають задуму композиції та добираються вигірні ракурси спостереження об'єктів (рис. 5).



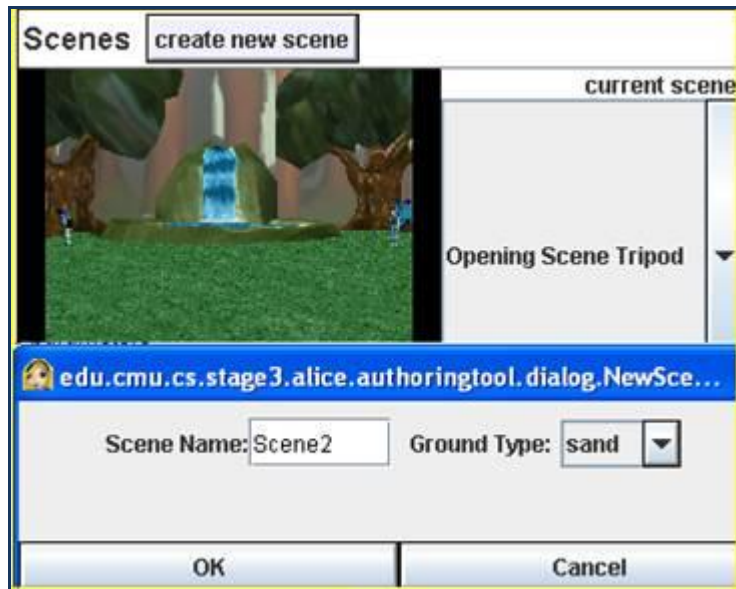
**Рис. 5.** Налаштування знімальних камер

5. Індивідуальне налаштування об'єктів. Для досягнення візуальної вірогідності і підвищення реалізму зображення, використовуються спеціальні налаштування об'єктів робочої сцени (рис. 6).



**Рис. 6.** Налаштування об'єктів

6. Моделювання сцен. На цьому етапі відбувається створення та редагування сцен для кожного епізоду фільму (рис. 7).



**Рис. 7.** Створення та редагування сцен

7. Анімація сцен полягає в автоматизації процесу формування послідовності сцен, що представляють окремі проміжні стадії руху того або іншого об'єкту або зміни його властивостей, таких як форма, колір (рис. 8).



**Рис. 8.** Формування послідовності сцени

8. Візуалізація сцен. Виконується візуалізація окремих об'єктів зображення сцени або послідовності об'єктів, що становлять анімацію (рис. 9).



Рис. 9. Візуалізація окремих об'єктів

«Alice» – програма, що є плодом інтелектуальної праці великого колективу розробників, які створювали і удосконалювали її не один рік. Вона наділена справді величезною кількістю параметрів, що допускає налаштування і забезпечує втілення практично будь-яких задумів користувача.

Відзначимо, що вивчення концепцій, конструкцій мов програмування досить складне для новачків будь-якого віку. Тому використання таких середовищ, як Alice, полегшує процес навчання студентів написання програм, вивчення синтаксису і т.п. У статті [12] автори наводять таксономію програмних середовищ і мов, які використовуються при навчанні новачків, а в статті [13] обговорюються програмний інструментарій, який повинен застосовуватися у ввідних курсах програмування.

У роботі [14] представлені результати спостережень за студентами, які не вивчали раніше програмування та пройшли навчання за допомогою Alice, в коледжі Ітаки та університеті Св. Йосифа. Дослідження підтвердили ефективність підходу в навчанні програмування за допомогою Alice.

**Інші можливості для сценарного підходу: середовище Scratch.** Існують і інші програмні середовища, аналогічні Alice. Наприклад, в медіа лабораторії Массачусетського технологічного інституту (MIT) створено середовище Scratch [15], [16], [17]. Scratch (з англійської мови можна перекласти як «начерк») являє собою комп'ютерну програму з простим графічним інтерфейсом, за допомогою якого учні можуть легко вивчити концепції об'єктно-орієнтованих мов програмування. На рис. 10 представлений приклад програми на Scratch. Відповідний сайт MIT [17] демонструє на безлічі прикладів чудові педагогічні можливості використання середовища Scratch.

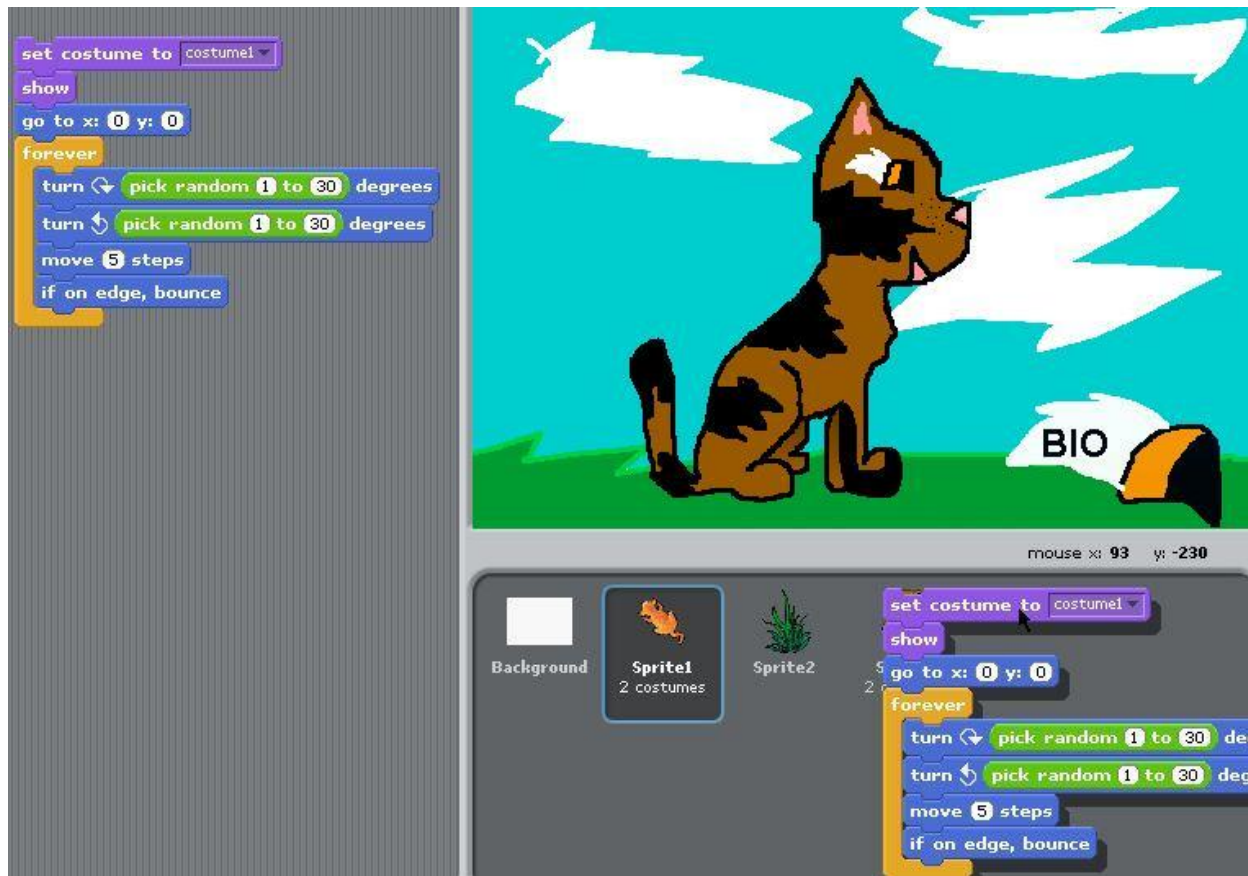


Рис. 10. Приклад програми в Scratch

\*\*\*

**Висновок.** Незважаючи на те, що описувані підходи апробовані тільки в навчанні університетських першокурсників США та України, нам здається цілком обґрунтованими сподівання на те, що застосування продуктів типу Alice і Scratch, а також розробка аналогічних українських версій для шкільної освіти, послужить справі модернізації навчання інформатики, пожевваленню цього навчання, різко підвищить інтерес старшокласників до активного і творчого використання комп'ютерів. Такий підхід переводить формальні алгоритмічні конструкції в проблематику, цікаву для початківця-програміста, робить програмування, як таке, природним і необтяжливим заняттям для мільйонів людей.

### Література

1. Ершов А.П. О человеческом и эстетическом факторах в программировании // История информатики в России: ученые и их школы / Сост.: В.И. Захаров, Р.И. Подловченко, Я.И. Фет. – М.: Наука, 2003. – С. 291-298.
2. Ершов А.П. Программирование – вторая грамотность // Архив академика А.П. Ершова [Электронный ресурс]. – Режим доступа: [http://ershov.iis.nsk.su/russian/second\\_literacy/article.html](http://ershov.iis.nsk.su/russian/second_literacy/article.html)
3. Жалдак М.І. Про деякі методичні аспекти навчання інформатики в школі і педагогічному університеті // Комп'ютерно-орієнтовані системи навчання / НПУ ім. М.П. Драгоманова. – 2005. – Вип. 9. – С. 3-14.
4. Інформатика. Програми для загальноосвітніх навчальних закладів / [ред. акад. М.І. Жалдака]. – Запоріжжя: Прем'єр, 2003. – 304 с.
5. Інформатика. Навчальна програма для учнів 9 класу загальноосвітніх навчальних закладів [Електронний ресурс]. – Режим доступа: [http://www.mon.gov.ua/education/average/new\\_pr/inf.doc](http://www.mon.gov.ua/education/average/new_pr/inf.doc)
6. Інформатика. Навчальна програма для учнів 10-12 класів загальноосвітніх навчальних закладів / Рівень стандарту [Електронний ресурс]. – Режим доступа: [http://www.mon.gov.ua/education/average/prog12/inf\\_standart.doc](http://www.mon.gov.ua/education/average/prog12/inf_standart.doc)
7. Інформатика. Навчальна програма для учнів 10-12 класів загальноосвітніх навчальних закладів / Академічний рівень [Електронний ресурс]. – Режим доступа: [http://www.mon.gov.ua/education/average/prog12/inf\\_academ.doc](http://www.mon.gov.ua/education/average/prog12/inf_academ.doc)
8. Сейдаметова З.С., Меджитова Л.М. Моделі навчання основ програмування на молодших курсах комп'ютерних спеціальностей університетів // Науковий часопис НПУ ім. М.П. Драгоманова. Серія №2. Комп'ютерно-орієнтовані системи навчання. Зб.наукових праць / Ред.рада. – К.: НПУ ім. М.П. Драгоманова, 2009. № 7(14). – С. 103–107.
9. Dann W.P., Cooper S., Pausch R. Learning To Program with Alice / 2nd edition. – NY: Prentice Hall, 2009. – 384 p.
10. Alice.org / Educational software that teaches students computer programming in a 3D



environment [Электронный ресурс]. – Режим доступа: <http://www.alice.org/>

11. Kelleher C., Pausch R. Using storytelling to motivate programming // *Communications of the ACM*, v. 50, # 7, 2007. – NY.: ACM. – Pp. 58 – 64.

12. Kelleher C., Pausch R. Lowering the Barriers to Programming: A Taxonomy of Programming Environments and Languages for Novice Programmers // *ACM Computing Surveys*, v. 37, # 2, 2005. – Pp. 88–137.

13. Powers K., Gross P., Cooper S., McNally M., Goldman K.J., Proulx V., Carlisle M. Tools for teaching introductory programming: what works? // *Proceedings of the 37th SIGCSE technical symposium on Computer science education*, 2006. – Houston, Texas, USA. – Pp. 560-561.

14. Moskal M., Lurie D., Cooper S. Evaluating the Effectiveness of a New Instructional Approach // *Proceedings SIGCSE Conference*, 2004. – Norfolk, VA, USA. – Pp. 75-79.

15. Resnick M., Maloney J., Monroy-Hernandez A., Rusk N., Eastmond E., Brennan K., Millner A., Rosenbaum E., Silver J., Silverman B., Kafai Y. Scratch: programming for all // *Communications of the ACM*, v. 52, # 11, 2009. – NY.: ACM. – Pp. 60-67.

16. Malan D.J., Leitner H.H. Scratch for budding computer scientists // *Proceedings of the 38th SIGCSE technical symposium on Computer science education*, 2007. – Covington, Kentucky, USA. – Pp. 223–227.

17. Scratch: a programming language for everyone / [Электронный ресурс]. – Режим доступа: <http://scratch.mit.edu/>