

ДІЛОВА ГРА ЯК МЕТОД АКТИВНОГО НАВЧАННЯ НА УРОКАХ ІНФОРМАТИКИ

Забарна А. П.

Теорія проблемного навчання (про яку йшлося у нашій статті «Проблемне навчання інформатики у профільній школі») може з однаковим успіхом застосовуватись як в основній, так і в старшій школі. Однак навчальна діяльність учнів у профільній школі має свою специфіку і в деякій мірі відрізняється від навчальної діяльності в основній школі (про мету і зміст профільного навчання говорилось вище). Основним навчальним завданням учнів середніх класів є засвоєння основ наук. При цьому головну роль відіграють широкі пізнавальні та навчально-пізнавальні мотиви. На відміну від цього педагогічний процес у старшій школі має вже деяку професійну спрямованість, передбачає вихід за межі навчальної діяльності і пізнавальної мотивації, актуалізацію важливих для майбутнього спеціаліста професійних мотивів.

Як відомо, профільне навчання передбачає створення умов для навчання старшокласників відповідно до їхнього професійного самовизначення. Отже, навчальна діяльність у профільній школі повинна дещо трансформуватись і бути не просто процесом передавання навчального матеріалу від учителя учневі, а простором, у якому школярі діють у контексті майбутньої професії. Для реалізації такого підходу до навчальної діяльності у профільній школі, наближення навчання школярів старшої школи до змісту майбутньої професії можна рекомендувати впроваджувати в педагогічну практику профільного навчання середньої школи концепцію контекстного навчання, запропоновану А. А. Вербицьким [4]. Ця концепція розроблена для підвищення ефективності навчання у вищій школі, проте деякі елементи її можуть з успіхом застосовуватись і в середній школі.

Контекстне навчання передбачає предметне та соціальне моделювання майбутньої професійної діяльності спеціаліста. Найбільш адекватними для цього є активні форми і методи навчання: аналіз конкретних ситуацій, розв'язування професійних задач, проблемні методи, ділові та рольові ігри, науково-дослідницька робота, стажування. З метою організації професійно-спрямованого навчання інформатики в профільній школі нами були розроблені практичні роботи на екологічну тематику [5–7]. Організована за теорією контекстного навчання діяльність набуває для старшокласників особистісного значення, оскільки в ній проглядаються ознаки майбутньої професії. Таким чином, поняття «професійний контекст» виступає смислоутворюючою категорією, що забезпечує особистісне включення старшокласників до процесу пізнання і професійного становлення.

Одним із методів активного навчання є ділова гра. У педагогічній літературі зустрічаються декілька означень поняття «ділова гра» [1; 2; 3; 9].

Проте аналіз різних її трактувань дає можливість зробити висновок, що головною особливістю ділової гри є наявність ситуації або імітаційної моделі, яка є

технологією виконання певних робіт (складання договору, розробка плану дій, розробка плану роботи, розробка програмного забезпечення тощо). Для ділової гри характерним є наявність:

- імітаційної моделі професійної діяльності та виробничих відносин;
- проблемної ситуації;
- ролей;
- рольових цілей та спільної мети всього колективу;
- взаємодія учасників гри, які виконують ті або інші ролі;
- колективної діяльності;
- ланцюжка рішень.

Розглянемо детальніше всі ці компоненти.

У будь-якій діловій грі імітується діяльність певної організації, підприємства або його підрозділу, наприклад, конструкторського бюро, будівельної фірми тощо. Можуть імітуватися події, конкретна діяльність людей (нарада, обговорення планів, проведення бесіди). У ділових іграх для школярів імітаційна модель представляється у спрощеному вигляді, без деталізації подробиць, які важливі для професіоналів. Це робиться з метою збереження уваги учнів, щоб вони не відволікалися на несуттєві питання. Модель імітації є стартовим моментом у конструюванні гри.

Після того, як модель визначена, ставиться проблемна ситуація, яка буде розв'язуватися у грі.

У будь-яких виробничих відносинах задіяні люди тих або інших спеціальностей. Тому вводять ролі спеціалістів, наприклад, бухгалтера, програміста, інженера-хіміка і т. д. Для кожної конкретної ролі існують ігрові цілі: що саме той або інший спеціаліст повинен виконувати в процесі гри та чого досягнути. У свою чергу в усього ігрового колективу також є мета — це спільна дидактична ціль, яка відповідає меті всієї гри. Ці дві цілі повинні бути врівноважені.

Ділова гра має декілька етапів: на початку гри ставиться проблемна ситуація, потім вона розбивається на підзадачі, кожна з яких розв'язується на одному з етапів, останній етап гри присвячений розв'язанню початкової проблеми. Тим самим протягом гри утворюється ланцюжок рішень.

Ділові ігри допомагають наблизити навчальний процес до реального життя. Проте це можливо за умови, якщо ці ігри моделюють реальні ситуації, а не схеми з підручника. Ефект від навчання за методом ділових ігор підсилюється також за рахунок перевтілення учасників гри в конкретних дійових осіб. Ділова гра породжує потужне психологічне поле, яке залучає до діяльності всіх учасників, викликаючи великий емоційний підйом. Гра дає можливість кожному її учаснику проявити творчі здібності, що є задоволенням потреб у саморе-



алізації; дає можливість позмагатися, підтвердити або змінити статус у групі.

У результаті проведення гри учні отримують:

- розуміння та засвоєння матеріалу підвищеної складності;
- розвиток умінь та навичок роботи в команді, прийняття рішень, здібностей контактувати та слухати інших, риторичних здібностей, лідерських якостей;
- розвиток продуктивного мислення, спостережливості, пам'яті, творчих здібностей, наполегливості, уміння доводити та відстоювати свою точку зору, організаторських здібностей.

Наведемо приклад використання методу навчання «Ділова гра» на уроках інформатики в профільній школі [8].

Тема уроку: Допоміжні алгоритми. Створення програм з використанням підпрограми-функції

I. Мотивація навчальної діяльності учнів.

Учні пригадують рольову гру, що її проводили на уроці під час вивчення теми «Етапи розв'язування задачі». Учні згадують, які ролі вони виконували, яку вони розв'язували задачу, яких висновків дійшли на уроці. Для ознайомлення з поняттям «допоміжний алгоритм» учитель повідомляє теоретичні відомості з нового матеріалу та знову пропонує розіграти рольову гру.

II. Оголошення, представлення теми та очікуваних навчальних результатів

Після цього уроку учні зможуть:

- пояснювати призначення допоміжних алгоритмів;
- пояснювати формат запису та виклику підпрограми-функції;
- називати основні кроки розробки програм з допоміжними алгоритмами;
- набути навичок розігрування ролей.

III. Надання необхідної інформації

Підпрограми призначені для реалізації алгоритмів опрацювання окремих частин деякої складної задачі. Вони дають змогу реалізувати концепцію структурного програмування, суть якого полягає у розкладанні складної задачі на послідовність простих підзадач і у складанні для алгоритмів розв'язування кожної підзадачі відповідних підпрограм. Розрізняють два види підпрограм — підпрограми-процедури та підпрограми-функції. Підпрограми поділяються на *стандартні* та *підпрограми користувача*. Стандартні підпрограми створювати не потрібно — вони містяться у стандартних модулях System, Crt, Dos, Graph тощо. Стандартні підпрограми приєднуються до тексту основної програми за допомогою розділу uses. **Підпрограма користувача** — це поійменована група команд, яку створюють й описують в основній (головній) програмі у розділах **procedure** або **function** і до якої звертаються з будь-якого місця програми необхідну кількість разів.

Підпрограма-функція може повертати у місце виклику лише один результат простого стандартного типу.

Загальний опис функції:

```
function <назва>(<список формальних параметрів>):
<тип функції>; <розділи описів і оголошень функції>;
begin
<розділ команд функції, де має бути така команда:
назва:=вираз>
end;
```

Тут, у списку **формальних** параметрів, перераховують змінні разом із зазначенням їхніх типів.

У розділі команд функції має бути команда присвоєння значення деякого виразу назві функції. Результат функції повертається в основну програму через її назву (як і випадку використання стандартних функцій таких, як sin, cos). Виклик функції здійснюється лише з виразів (тобто виклик функції повинен бути праворуч оператора присвоєння):

```
<змінна>:= <назва> (<список фактичних параметрів>)...
```

Приклад. Створимо функцію для обчислення $\text{tg}(x)$ та обчислимо значення виразу: $\lg(x)+\text{ctg}(x)+\text{tg}^2(x)$.

```
program Myfunc;
uses Crt;
var x,y:real;
function tg(x:real):real;
begin tg:=sin(x)/cos(x)
end;
begin clrscr;
  writeln('Введіть x'); readln(x);
  y:=tg(x)+1/tg(x)+sqr(tg(x));
  writeln('y=', y:5:2); readln
end.
```

IV. Інтерактивна вправа

Учитель описує учням ситуацію, подібної до рольової гри, яка проводилася на уроці з вивчення теми «Етапи розв'язування задачі». Проте деякі умови гри змінюються.

Є фірма-розробник програмного забезпечення. Нехай ця фірма називається «Інтелект» (можна запропонувати учням дати назву фірмі). У цю фірму приходить екодизайнер міста, який працює над проектом озеленення міста та впорядкування місць відпочинку. У місті вже є багато скверів, проте вони недостатньо облаштовані для відпочинку населення. Для виділення коштів на впорядкування скверів треба знати їхню загальну площу. Треба написати програму, яка дозволить обчислити площу всіх скверів міста за умови, що деякі сквери мають форму прямокутника, деякі — форму трикутника, а деякі — форму трапеції. Учитель пропонує учням зімітувати роботу фірми з виконання замовлення екодизайнера. Для цього вчитель пропонує учням розіграти ролі (бажано ролі розподілити іншим чином, ніж у попередній рольовій грі).

Екодизайнер (1 учень) — замовник роботи.

Постановники задачі першого бюро (2 учні), *постановники задачі другого бюро* (2 учні), *постановники задачі третього бюро* (2 учні), *постановники задачі головного бюро* (2 учні) — формулюють постановку задачі для програмістів своїх бюро.

Програмісти першого бюро (2 учні) — складають та перевіряють алгоритм обчислення площі трикутника, пишуть програму-функцію **S_tryk** обчислення площі трикутника за трьома сторонами.

Програмісти другого бюро (2 учні) — складають та перевіряють алгоритм обчислення площі прямокутника, пишуть програму-функцію **S_pryam** обчислення площі прямокутника.

Програмісти третього бюро (2 учні) — складають та перевіряють алгоритм обчислення площі трапеції, пишуть програму-функцію **S_trap** обчислення площі трапеції.

Програмісти головного бюро (2 учні) — складають тіло головної програми на обчислення площі всіх скверів.

Програміст-експерт (1 учень) — об'єднує в одну програму результати роботи всіх чотирьох бюро.

Учні отримують сценарій гри з описом своїх дій, готуються до розігрування ролей, тренуються.

Орієнтовний сценарій гри

Екодизайнер приходить у фірму «Інтелект» та повідомляє свою проблему, викладає суть роботи *постановникам задачі*.

Постановники вивчають задачу, формулюють для чотирьох груп програмістів математичну модель задачі, визначають вхідні та вихідні дані (чітко з'ясовують, що дано і що треба знайти), встановлюють характеристики (типи) вхідних та вихідних даних.

Програмісти першого бюро — складають та перевіряють алгоритм обчислення площі трикутника, пишуть програму-функцію **S_tryk** обчислення площі трикутника за трьома сторонами, користуючись фрагментом цієї програми, що його треба допрацювати (замість знака питання «?» вставити необхідний текст):

```
function S_tryk(x,y,z:?):?;
var p:real;
begin
p:=(x+y+z)/?; S_tryk:=?(p*(p-x)*(p-y)*(p-z));
end;
```

Програмісти другого бюро — складають та перевіряють алгоритм обчислення площі прямокутника, пишуть програму-функцію **S_pryam** обчислення площі прямокутника, користуючись фрагментом цієї програми, що його треба допрацювати (замість знака питання «?» вставити необхідний текст):

```
function S_pryam(x,y:?):?;
begin S_pryam:=x*y;
end;
```

Програмісти третього бюро — складають та перевіряють алгоритм обчислення площі трапеції, пишуть програму-функцію **S_trap** обчислення площі трапеції, користуючись фрагментом цієї програми, що його треба допрацювати (замість знака питання „?» вставити необхідний текст):

```
function S_trap(x,y,z:?):?;
begin S_trap:=(x+y)?2*z;
end;
```

Програмісти головного бюро — знаючи назви підпрограм-функцій та їхні списки формальних параметрів, складають тіло головної програми на обчислення площі всіх скверів.

Коли всі бюро програмістів закінчать свою роботу, *програміст-експерт* об'єднує в одну програму результати роботи всіх чотирьох бюро:

```
uses crt;
var i, num:word;
s,a,b,c,h:real;
function S_tryk(x,y,z:real):real;
var p:real;
begin p:=(x+y+z)/2; S_tryk:=sqrt(p*(p-x)*(p-y)*(p-z));
end;
function S_pryam(x,y:real):real;
begin S_pryam:=x*y;
end;
function S_trap(x,y,z:real):real;
begin S_trap:=(x+y)/2*z;
end;
begin
```

```
write('Скверів у вигляді трикутника:');
readln(num); s:=0; {загальна площа}
for i:=1 to num do
begin readln(a,b,c); {введення сторін трикутника}
s:=s+S_tryk(a,b,c);
end;
write('Скверів у вигляді прямокутника:'); readln(num);
for i:=1 to num do
begin readln(a,b); {введення сторін прямокутника}
s:=s+S_pryam(a,b);
end;
write('Скверів у вигляді трапеції:'); readln(num);
for i:=1 to num do
begin readln(a,b,h); {введення довжини основ та висоти трапеції}
s:=s+S_trap(a,b,h);
end;
writeln('Площа всіх скверів дорівнює',s:6:2,' кв.м');
readln;
end.
```

Коли програма написана та перевірена, тоді правильність її роботи перевіряють *постановник*: чи відповідає складений алгоритм вимогам постановки задачі.

Екодизайнер отримує готову програму для постійного користування (в експлуатацію). *Зауваження*: під час роботи інших учасників рольової гри екодизайнер розробляє критерії перевірки правильності виконання замовлення.

Учитель у формі бесіди обговорює з учнями питання «Чи можна було обійтися у даній ситуації без використання допоміжних алгоритмів? Коли доцільно використовувати допоміжні алгоритми?». Учитель підводить учнів до думки, що у тих випадках, коли треба розв'язати велику (складну) задачу, тоді вона розбивається на підзадачі, які розв'язуються у допоміжних алгоритмах. Це зручно робити, бо у цьому випадку можна налагодити кожен таку програму окремо, а вже згодом об'єднувати всі програми у головній програмі. Учитель пропонує учням відповіді на такі запитання:

1. Фахівцем якої справи (з перелічених) вам би хотілося бути?
2. Як ви почувалися у своїй ролі?
3. Які ще задачі з використанням допоміжних алгоритмів можна запропонувати працівникам фірми «Інтелект»? На які підзадачі будуть розбиті ці задачі?

V. Підбиття підсумків, оцінювання результатів уроку

Учитель проводить опитування:

1. Як ви гадаєте, що треба враховувати розробникам допоміжних алгоритмів, щоб зібрані в одній головній програмі вони правильно працювали?
2. Висловіть вашу думку, чи залежить якість програм від професійного рівня фахівців? Яких саме фахівців?
3. Як ви гадаєте, чи досягли ми мети, яку поставили на початку уроку?
4. Яких нових знань та вмінь ви набули на цьому уроці?

VI. Повідомлення домашнього завдання

Наведена розробка уроку є прикладом того, як можна застосувати методи активного навчання, зокрема метод «Ділова гра», на уроках інформатики в профільній школі.

Література

1. Бабурин В. Л. Деловые игры по экономической географии. — М: Просвещение — АО «Учебная книга», 1995. — 158 с.
2. Бадаев А. А. Активные методы обучения. — М.: Профиздат, 1986. — 127 с.
3. Бельчиков Я. М, Бирштейн М. М. Деловые игры. — Рига: Авотс, 1989. — 218 с.
4. Вербицкий А. А. Активное обучение в высшей школе: контекстный подход. — М.: Высшая школа, 1991. — 206 с.
5. Забарна А. П. Досвід проведення практичних робіт з інформатики з урахуванням профілізації навчального закладу // Комп'ютер у школі та сім'ї. — 2006. — №2. — С. 35–38.
6. Забарна А. П. Інформатика в загальноосвітніх навчальних закладах природничо-математичного напрямку профілізації // Науковий часопис НПУ ім. М. П. Драгоманова. Серія №2. Комп'ютерно-орієнтовані системи навчання: Зб. наук, праць / Редкол. — К.: НПУ ім. М. П. Драгоманова. — №3(10) — 2005. — С. 50–59.
7. Забарна А. П. Практичні роботи на екологічну тематику в середовищі MS EXCEL // Інформатика. — №13 (301), квітень 2005 р. — С. 15–17.
8. Забарна А. П. Основи алгоритмізації та програмування. Інтерактивні технології навчання на уроках. — Тернопіль: Мандрівець, 2006. — 96 с.
9. Платов В. Я. Деловые игры: разработка, организация, проведение. — М.: ИПО «Профиздат», 1991. — 152 с.



НАВЧАННЯ ОЛІМПІАДНОЇ ІНФОРМАТИКИ ЯК ОСОБИСТІСНА САМОРЕАЛІЗАЦІЯ УЧНЯ

Мельник В.І.

Напрями та особливості розвитку особистості учня у навчально-виховному процесі загальноосвітньої школи традиційно визначаються соціальними і матеріальними умовами життя суспільства. Особистісна самореалізація школяра нерозривно пов'язана з розвитком освіти, науки, промисловості.

Українська національна система освіти результатом шкільного навчання проголосила формування компетентної особистості. **Компетентність** — це характеристика особи, що формується у процесі навчання і вміщує знання, навички, ставлення, досвід діяльності і поведінкові моделі людини. Найважливішим моментом у викладанні предмету є поведінка учня в системі «людина-комп'ютер».

Компетентнісний підхід полягає у зміщенні акценту від накопичення знань до формування і розвитку в учнів здатності практично діяти в інформаційному полі та застосовувати індивідуальний навчальний досвід у професійній діяльності та соціальній практиці.

Олімпіадна інформатика — це особливий міжнародний рух людей, які прагнуть скласти найкращі алгоритми для розв'язання будь-якої проблеми правильно, швидко і «красиво». І мова програмування є тільки інструментом для реалізації ІДЕЇ.

Термін «самореалізація» (selfrealisation) вперше було наведено у «Словнику з філософії та психології», виданому в 1902 році. Найчастіше поняття «самореалізація» інтерпретується як «реалізація власного потенціалу».

Виробленню учнем особистого стилю навчання олімпіадної інформатиці ефективно сприяє робота в різновікових групах. Освітньою перевагою системи роботи у різновікових групах є її вмотивованість, що базується на комплексній процесуальній теорії мотивації Лаймана Портера і Едварда Лоулера, що включає елементи теорії очікувань і теорії справедливості [5] та спрямованість на самореалізацію (рис. 1).

Результативність системи роботи в різновікових групах, як освітньої технології доводять успішні виступи моїх учнів на Міжнародних олімпіадах з інформатики: 2006 р. — срібна медаль Сімоненка Владислава на Міжнародній учнівській олімпіаді з інформатики (м. Медіна, Мексика) серед 282 учасників з 78 країн; 2007 р. — бронзові медалі Сімоненка Руслана та Джумані-

язова Рустама на Міжнародній учнівській олімпіаді з інформатики (м. Загреб, Хорватія) серед 277 учасників з 77 країн; 2008 р. — срібна медаль Сімоненка Руслана на Міжнародній учнівській олімпіаді з інформатики (м. Каїр, Єгипет) серед 288 учасників з 82 країн; 2009 р. — срібна медаль Паламарчука Степана на Міжнародній олімпіаді з інформатики (м. Пловдив, Болгарія).

Робота учнів у різновікових групах проводиться на засадах:

1. Прискорення (інтенсифікація навчання за спецпрограмами для кожного).
2. Поглиблення (вивчення окремих дисциплін за об'єктивними потребами учня).
3. Збагачення (спеціальні інтелектуальні тренінги розвитку тих чи інших здібностей дитини, участь у міжнародних проектах, Інтернет-олімпіадах).
4. Проблематизації (стимулювання особистісного розвитку учня у напрямі розширення компетентності).

Використання різних Інтернет-ресурсів (табл. 1) дозволяє оптимізувати діяльність учнів під час підготовки до олімпіад з інформатики не тільки у класі, але й удома, і є однією з ефективних форм практичної реалізації пріоритетних завдань пошуку, навчання. Причому пропонує учням Інтернет-джерела з англійськомовним, російськомовним та українськомовним інтерфейсом, що допомагає у формуванні не тільки інформативної, а й комунікативної ключових компетентностей.

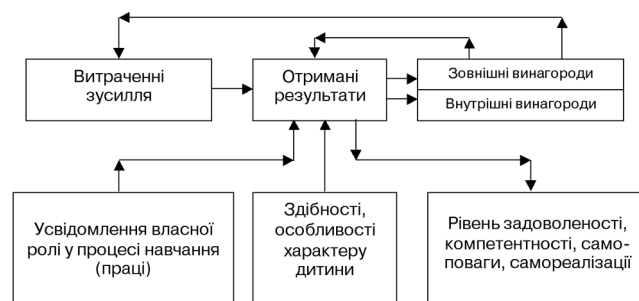


Рис. 1