

## **Проблеми формування алгоритмічної культури майбутніх вчителів інформатики**

Навряд чи потрібно згадувати про ту роль, яку відіграють інформаційні технології в сучасному суспільстві. Не секрет також, що фундамент цих технологій забезпечується інформатикою, і насамперед, її найважливішою складовою частиною – алгоритмікою. На жаль, в Україні ще не сформоване адекватне відношення суспільства до цієї науки. Це проявляється й у залишковому принципі планування відповідної дисципліни в більшості шкіл, й у розповсюдженому уявленні про неї, як про суто допоміжну, "проофісну" дисципліну. Як правило, ігнорується розвиваючий, творчий та загальнокультурний потенціал інформатики. З іншого боку панує відношення до фундаментальних її основ як до чогось надміру зарозумілого і непотрібного в повсякденному житті, чогось лякаючого своєю складністю, недоступністю, незвичайністю.

Уряд України та Міністерство науки і освіти звернули увагу на цей стан речей і прийняли ряд відомих програм і рішень, направлених на його поліпшення. Але без підтримки широкого кола вчителів, що можуть донести прогресивні рішення керівництва системи освіти до кожної дитини та її батьків, ці рішення залишаються не реалізованими, сьогоденній стан інформаційної освіти навіть не посередній, а відверто низький.

Фігура вчителя інформатики є центральною і в процесі інформатизації школи, і в організації "лікбезу" силами самої школи, і у формуванні інформаційної культури в учнів.

На останній Всеукраїнській науково-методичній конференції "Нові інформаційні технології навчання в учбових закладах України", що традиційно пройшла в Одесі у вересні 2003 року, значна увага приділялась питанням інформатизації школи та педвузу. Як на пленарних, так і на секційних засіданнях, зокрема, на секції 5 "Комп'ютерно-орієнтовані системи навчання у школі та педвузі" (керівники секції М. І. Жалдак та автор цієї статті), відмічалось, що просте насичення школи комп'ютерами ще далеко не вирішує проблеми інформатизації. Нагальною необхідністю є передусім різке підвищення культури володіння цією технікою та інформаційної культури в цілому, широке використання сучасних інформаційних технологій в навчальному процесі при вивченні всіх предметів. Рішення конференції прямо підкреслюють невід'ємність цих проблем від проблем інформатизації навчального процесу, всієї системи освіти, більше того – їх першочерговість.

Відтак, задача формування інформаційної культури майбутніх вчителів всіх предметів тим більш першочергова.

Первісно, поняття *культура* означало культуру землеробства, під якою розуміли систему правил, традицій, пов'язаних з цілеспрямованим управлінням процесами вирощування рослин та збирання врожаїв. Потім це поняття охопило й інші сфери життя та діяльності людини і суспільства, створюваних матеріальних та духовних цінностей. В процесі розвитку людства виділились і відособились окремі субкультури. Однак, будь-яка культурна діяльність здійснюється у відповідності з певними ритуалами, правилами, або супроводжується, опосередковується ними. Діяльність, пов'язана зі створенням та використанням алгоритмів, отримала особливо бурхливий розвиток у зв'язку з необхідністю програмування обчислювальних машин – комп'ютерів. Поняття *алгоритмічна культура* має свої родові корені у науковій літературі, починаючи з 70-х років ХХ сторіччя, коли у працях Г. Г. Вороб'йова, С. В. Михайліді, Н. М. Розенберга, В. М. Монахова, А. П. Єршова [1] вводяться поняття *інформаційна* та *комп'ютерна культура*.

З моменту її виникнення у вітчизняній інформатиці найпильнішу увагу звертали на задачу формування особливого алгоритмічного мислення, необхідного людині для адекватного використання з комп'ютерних систем.

У відповідності до загальноприйнятого визначення культури взагалі можна констатувати, що алгоритмічна культура людини відображає рівень розвитку мислення, зокрема логічного в цілому, типів і форм організації використання комп'ютера, матеріалізованих у створених алгоритмах, програмах.

Під *алгоритмічною культурою людини* будемо розуміти культуру *розробки, використання та вибору* алгоритму для здійснення певної діяльності або досягнення конкретної мети (вирішення задачі) ефективним, оптимальним у деякому розумінні, чином.

Розглянемо три головні компоненти алгоритмічної культури, що виділені в даному визначенні: розробку, використання та вибір алгоритму. Щоб підкреслити саме культурну їх сутність, наведемо аналогію з відповідними компонентами в інших загальноновизначаних явищах культури. Візьмемо, наприклад, літературу. Маємо відповідні компоненти: культуру написання книг, культуру читання книг та культуру вибору книг для читання. Все це безсумнівно цінні та загальноновизнані елементи культури.

У вузькому розумінні під *алгоритмічною культурою* слід дами на увазі лише перший з її компонентів, а відповідну науку можна назвати *алгоритмікою*. Взагалі ж, розробка, або проектування, алгоритмів (тісно пов'язана сьогодні з проектуванням інтерфейсів) є також предметом *програмування*, як науки, мистецтва і технології [3]. Як відомо, програмування тісно пов'язане з алгоритмікою за допомогою *мов програмування*, або *алгоритмічної мови*.

Як *наука* програмування розглядає свою типізацію та класифікацію алгоритмів, виділяючи при цьому елементарні алгоритми-"цеглини", методи розробки програм для поєднання "цеглин" у цілісну "споруду", способи доведення правильності програм, аналізує синтаксичні та семантичні аспекти алгоритмічних мов. В основі цієї науки лежить математична логіка та теорія алгоритмів, що пропонує

своєрідний постулат віри – будь-який алгоритм може бути реалізований в одній з відомих математичних алгоритмічних систем. Однак проблема неповноти, відкрита Гьоделем, ставить під питання закінченість та надійність створеного математичного фундаменту. Можна також погодитися з висновком про те, що прикладна математика, як математична основа програмування, є одним із найскладніших розділів сучасної математики. Саме в програмуванні людина зустрічається з алгоритмами безпрецедентної довжини і складності, що перевершують можливості звичайного людського сприйняття і не дозволяють обмежитися суто науковими методами і підходами.

Як *мистецтво*, програмування являє собою процес індивідуальної та колективної творчості. Його продукти призначені для паралельного сприйняття як штучним, так і природним (людським) інтелектом, а саме, як безпосередніми виконавцями-комп'ютерами, так і іншими людьми-програмістами. Тут доводиться мати справу з новою психологією - психологією програмування [4], при цьому, небувалого розвитку набуває колективна творчість, з'являються нові її форми, наприклад, екстремальне програмування. При збереженні традиційних якісних критеріїв оцінки програм, як творів мистецтва, з позицій гармонійності, зрозумілості, простоти, лаконічності, краси, вперше в історії мистецтва з'являється принципова можливість кількісної їх оцінки. Наприклад, це можливо шляхом обчислення параметра *досконалості* програми [5]. Це можна розцінювати як один з результатів порівняння двох типів інтелекту, як оцінку творів природного інтелекту штучним. На стику науки і мистецтва можна розташувати також *філософію програмування* (див., наприклад, [6]).

Як *технологія*, програмування є основою найпотужнішої галузі сучасного виробництва, до якої залучені мільярди капітали та мільйони людей. Серед найбільш відомих технологій програмування та тих, що вивчаються — структурне, модульне та об'єктно-орієнтоване програмування. Небачені масштаби та темпи зростання виробництва програм, при тісному зв'язку з виробництвом комп'ютерної техніки регулюються законом Мура і водночас характеризуються постійним станом кризи. Зокрема, це проявляється у такому негативному явищі, як засилля великих програм, та багатьох інших. Боротьба з подібними негативними явищами виходить далеко за рамки технології і програмування, перетворюючись в задачу суспільства і людства в цілому. Вирішення подібних проблем слід шукати на шляхах культурного розвитку. Сьогодні культура програмування тісно пов'язана також з питанням авторського права та деякими іншими правовими питаннями.

Відволікаючись від власне програмування, розглянемо компоненти, що створюють поняття *алгоритмічна культура в широкому його сенсі*. *Культура використання* алгоритмів має дві головні сторони – *культуру виконання* та власне *культуру користування*. Користувач комп'ютера повинен володіти також і навичками виконання алгоритмів, без чого неможливе свідоме користування комп'ютерною системою та її розуміння на будь-якому рівні. Особливо це стосується професійних операторів.

*Культура вибору* потребує обов'язкової наявності досить широких знань про існуючі алгоритми та програми, їх типи та призначення, а також знань основ тих предметних галузей для розв'язування задач, з яких розроблені програми. Крім того, необхідно володіння сучасними методами пошуку, зокрема, пошуковими системами Інтернет, навичками мережевого спілкування, тощо.

Подібно до структури культури взагалі, в рамках алгоритмічної культури існують цікаві та важливі субкультури. Виділимо мережеву субкультуру, мовні субкультури (Pascal, C, Java), операційні субкультури (Windows, Unix, Linux). Між деякими субкультурами існують серйозні протиріччя, у тому числі на психологічному або економічному ґрунті, що призводять іноді до тривалих протистоянь та конфліктів. Водночас існує прогресивна тенденція взаємопроникнення та взаємообміну субкультур. У відповідності з класичними уявленнями, культура проявляється у таких формах, як релігія, мистецтво, філософія, право. Найбільш розвинуті вияви цих форм - у сучасній мережевій субкультурі.

З вищенаведеного випливає, що всі основні функції культури – інформативна, регулятивна і комунікативна – повноцінно конкретизуються та реалізуються у сфері алгоритмічної культури. Класичне (бо походить ще від античності) уявлення про культуру, як про гармонію, також цілком відповідає інтуїтивному уявленню професіоналів у цій галузі про алгоритмічну культуру. Бо гармонійність алгоритму і програми проявляються в їх високій ефективності, якості, зручності.

Однією з найважливіших задач сучасної школи всіх рівнів та ступенів залишається формування культурної, всебічно розвиненої людини. Завдання формування алгоритмічної культури широких верств суспільства, а перед усім вчителів та учнів, залишається першочерговою.

Однак на сьогодні саме тут виникає низка проблем. *Перша проблема* – це проблема *недостатнього рівня підготовки з інформатики тих абітурієнтів*, що поступають на відповідні спеціальності педвузів. По-перше, практично у всіх педвузах відсутні вступні іспити або співбесіди з інформатики. По-друге, більшість абітурієнтів приходять з сільських шкіл, де комп'ютерів практично немає, а тому і інформатика дається в мінімальному обсязі. Але навіть, якщо комп'ютери є, наприклад отримані за Державною Програмою інформатизації загальноосвітніх навчальних закладів, комп'ютеризації сільських шкіл, то немає кваліфікованого вчителя.

З першою тісно пов'язана *друга проблема*. Проблема *нерівномірності рівнів підготовки абітурієнтів*. Бо поступають також випускники шкіл, де з інформатикою все гаразд, або вони займалися самостійно, чи додатково. І відсоток таких абітурієнтів зростає.

Ці проблеми педвузи вирішували завжди, тим чи іншим чином організовуючи "лікбез", та враховуючи початковий рівень студентів у навчальних програмах. Однак більш послідовне рішення дає введення у першому семестрі додаткового навчального курсу "Вступ до інформатики" або "Шкільний курс інформатики". Направленість цього курсу має бути пропедевтично-узгальною, щоб, з одного боку, підтягнути слабких, а з іншого боку, не розхолодити та дати корисний матеріал

також і сильним. Такий курс читається автором на протязі вже п'яти років, і результати можуть бути визнані добрими. Але на жаль, не для всіх спеціальностей вдається знайти години на додатковий курс, особливо у першому семестрі.

І тут ми стикаємося з *третьою проблемою*. Це проблема *подвійних спеціальностей*. Річ у тім, що у більшості шкіл недостатньо годин з інформатики для утримання бодай однієї повної ставки вчителя інформатики. Тому він повинен підробляти математиком, або фізиком, або економістом, або ще кимось. Тому ж він, як правило, отримав у вузі не повноцінну спеціальність інформатика, а так звану другу, де і кількість годин втричі менша, та й ставлення до неї не відповідне. Зокрема, ввести згаданий додатковий курс вдається лише там, де інформатика – перша спеціальність.

Третя проблема посилює першу, одночасно породжуючи *четверту*. Проблему *гуманітарної орієнтованості абітурієнтів і студентів*. Для педвузу це природно, для студента з нематематичною першою спеціальністю це більш, ніж природно. Однак, крім інформатики, лише при навчанні шкільної математики можна сформувати необхідні алгоритмічні знання та навички.

Якщо ж студент не підготований ні до інформатики (перша проблема), ні до математики, то єдиний вихід – якомога більше гуманітаризувати курс інформатики для студентів подібних спеціальностей. І тут точкою опори може стати поняття алгоритмічної культури, точніше, підхід до складних проблем алгоритмізації саме як до явища культури. Це зміщує акценти в курсі інформатики для гуманітаріїв. Стосовно змістової наповненості подібного курсу інформатики плідною видається ідея ідея систематичного співставлення явищ, форм та методів алгоритмічної культури з аналогами у загальновідомих культурних проявах. Наприклад, з мистецтвом – музикою, тощо. Можна також навести приблизні назви додаткових інформатичних курсів спрямованості, що видаються перспективними: "Історія та філософія програмування", "Краса у програмуванні", "Право та релігія віртуального світу", "Проблеми духа та свідомості у штучному інтелекті" (останній курс читався 2 роки в одній з Академій міста Одеси).

*П'ята проблема* пов'язана з тим, що вчитель інформатики *одночасно є* (на суспільних, звичайно, засадах) і *інженером з ремонту комп'ютерів*, і проектувальником мереж, і відповідальним за безперебійне функціонування Windows та Microsoft Office у масштабі всієї школи, так ще й антивірусним доктором і єдиним у полі воїном з комп'ютерними гравцями і малолітніми хакерами. Доводиться виділяти лічені години у навчальних планах і на ці цілі. Але насправді єдина серйозна надія у вирішенні цієї проблеми – на самого студента-інформатика, його самостійну роботу і консультативну допомогу викладачів.

Охоплене коло проблем – це здебільшого *зовнішні проблеми*. Ми лише підходимо до *внутрішніх проблем*, пов'язаних з різноманітністю і великою кількістю навіть самих основних алгоритмів та їх типів, алгоритмічних мов, операційних систем і платформ, інформаційних систем та баз даних, мереж, взагалі - програмного забезпечення різноманітного призначення, необхідністю багатьох допоміжних математичних курсів, тощо. Слід також підкреслити, що не всі компоненти алгоритмічної культури в достатній мірі відображені у змісті навчальних курсів інформатичного циклу. Ми не торкалися і проблеми підручників і посібників з інформатики для педвузів.

#### ЛІТЕРАТУРА

1. Ершов А. П. Школьная информатика в СССР: от грамотности к культуре// Информатика и образование - 1987 - №6 - с. 3-11.
2. Малорян В. Л. Новый подход к тестированию и оценке интеллектуальных систем.- Нові інформаційні технології навчання в учбових закладах України. Зб. статей по проблемам шостої Української наук.-метод. конф.- Одеса, 1999, с.75.
3. Кнут Д. Искусство программирования для ЭВМ.- т.1. Основные алгоритмы.- М., Мир, 1976.
4. Шнейдерман Б. Психология программирования.- М., Радио и связь, 1984.
5. Холстед М. Х. Начала науки о программах.- М., Финансы и статистика, 1981.
6. Эккель Б. Философия Java.- СПб, Питер, 2001.