### Поняття змінної величини

#### 2.6.1. Змінні величини

Поняття змінної числової величини з'явилося в роботах Декарта в XVII ст. Пізніше з'ясувалося, що змінна величина може бути не лише числовою. У найширшому розумінні **змінна величина** – це загальне поняття реального чи уявного об'єкта (або його окремої характеристики), який може перебувати в різних станах. Змінні величини позначають *іменами*; наприклад, у другому законі Ньютона *a* *F*/*m* імена *m*, *a*, *F* позначають змінні – масу тіла, прискорення його руху й силу, що діє на нього. У програмуванні змінна є моделлю (зображенням) об'єкта в пам'яті комп'ютера.

**Змінна в машинній програмі** – це ділянка пам'яті, яка може мати різні стани й зображувати ними різні стани об'єкта.

Змінну в програмі мовою високого рівня позначає **ім'я**. Кожна ділянка пам'яті комп'ютера має **адресу** (номер першого байта ділянки), і цією адресою позначається в машинній програмі. Кажуть, що адреса *вказує*, або *посилається* на ділянку. Під час трансляції програми ім'я змінної перетворюється на адресу деякої ділянки пам'яті. Отже, вважатимемо, що під час виконання програми ім'я змінної *вказує* на ділянку пам'яті (рис. 2.1).

***Ім***

***'***

***я***

***у***

***програмі***

***Змінна***

##### Рис. 2.1. Змінна та її ім'я

Станам об'єкта, зображуваного змінною, відповідають її **значення** (числові та інші), що належать певному типу. Тип змінної та її ім'я задаються в **інструкції оголошення імені змінної**, що має вигляд ***ім'я-типу ім'я-змінної*;**

Після імені типу можна записати кілька імен змінних через кому. Інструкція оголошення імені змінної одночасно є **означенням змінної**, тобто задає ділянку пам'яті, на яку вказує ім'я змінної.

**Приклади.**

1. Інструкції **int numberOfStudents; double radius, circleLength;**

задають утворення ділянки пам'яті, яка ідентифікується іменем **numberOfStudents** і може зберігати цілі числа, і двох ділянок з іменами **radius** і **circleLength**; їх значеннями можуть бути дійсні числа.

1. Квадратне рівняння *ax*2+*bx*+*c* 0 визначається трьома дійсними коефіцієнтами *a*, *b*, *c*. Щоб зобразити рівняння, означимо три змінні: **double a, b, c;.** 

Імена змінних та інших об'єктів обирають так, щоб вони були пов'язані з об'єктами або характеристиками, які вони зображують, наприклад **temperature**, **radius**, **circleLength** (температура, радіус, довжина кола).

Завдяки змістовним іменам текст програми стає зрозумілішим.

Ім'ям змінної не може бути зарезервоване слово. Імена, означені в бібліотеках системи програмування, використовувати як імена змінних можна, але краще цього не робити.

Надалі будемо дотримуватися таких угод. Ім'я змінної починаємо з малої літери; якщо воно утворене з кількох слів, то всі слова, окрім першого, записуємо з великої літери, наприклад **circleLentgh**. Ця система запису нині є найпоширенішою у світі й через зовнішній вигляд імен має назву **"верблюжий запис"** (*camel notation*).

Якщо потрібна невелика й відома наперед кількість змінних, що позначають однотипні сутності, то в кінці імені можна дописати номер, наприклад **radius1**, **radius2**.

Існує давня традиція, за якою імена **i**, **j**, **k**, **m**, **n** дають змінним цілих типів. Найчастіше ці змінні позначають індекси (номери) або кількості.

Узагалі, змінна може бути складовою частиною іншої змінної. Тоді вона позначається не ім'ям, а деяким складнішим виразом. Однак поки що не будемо на цьому зосереджуватися.

#### Вправи

2.10. Написати означення змінних, що зображують: а) день, місяць і рік народження; б) точку площини; в) клітинку шахового поля; г) дріб; д) квадратне рівняння; е) прямокутник на площині.

2.11. Пояснити різницю між: а) **a** та **'a';** б) **a** та **"a"**.

##### 2.6.2. Уведення значень у змінні й виведення їх на екран

Є два способи надати змінній значення – увести із зовнішнього носія або присвоїти. У найпростішому випадку **інструкція введення** значення в змінну має вигляд **cin >> ім'я*-*змінної;**

і задає введення з клавіатури. Ім'я **cin,** як і **cout,** оголошене у файлі **iostream**. Виконуючи операцію введення, комп'ютер зупиняється й очікує на значення для змінної. У відповідь слід на клавіатурі набрати деяку послідовність символів, що зображує значення (ці символи з'являться на екрані), і натиснути на клавішу **Enter**. Після цього за введеною послідовністю (вхідною константою) створюється відповідне значення та присвоюється змінній. За стандартних налаштувань запис числа сприймається як десятковий.

Якщо натиснути **Enter**, не набравши нічого, окрім пробілів, то комп'ютер і надалі чекатиме. Узагалі, перед вхідною константою можна набрати довільну кількість порожніх символів, що відповідають клавішам пробілу, табуляції та **Enter**. За стандартних налаштувань усі порожні символи буде пропущено.

В інструкції введення можна записати кілька імен змінних, кожне після відповідного знака **>>**. За виконання такої інструкції треба набрати на клавіатурі відповідну кількість вхідних констант, відокремивши їх одним або кількома порожніми символами. **Приклади**

1. Нехай є змінна **char c;** і виконується інструкція **cin >> c;.** Щоб присвоїти змінній значення **'A'**, треба натиснути послідовно на клавіші **A** та **Enter**, а значення **'7'** – на клавіші **7** та **Enter**. Якщо перед клавішею **A** або **7** кілька разів натиснути на клавіші **Enter** або **Space**, то результат буде той самий.
2. Нехай змінні **double a, b, c;** зображують коефіцієнти квадратного рівняння. Увести в них дійсні значення можна за допомогою трьох окремих інструкцій: **cin >> a; cin >> b; cin >> c;**

Ті самі дії буде задано й однією інструкцією: **cin >> a >> b >> c;**

В обох ситуаціях на клавіатурі слід набрати три константи, натискаючи між ними на клавіші пробілу, табуляції або **Enter**. Виконання інструкцій закінчиться, коли після третьої константи буде натиснуто на **Enter**. 

Практично перед кожною інструкцією введення з клавіатури варто записати інструкцію виведення, яка запрошує до введення значень і вказує, скільки та яких типів.

**Приклад**. Уведення значень у дійсні змінні, що зображують коефіцієнти квадратного рівняння, можна оформити таким чином:

**cout << "Enter coefficients of quadratic equation "; cout << "(three real or integer numbers):\n";**

**cin >> a >> b >> c;**

Після появи цього запрошення курсор буде переведено в наступний рядок екрана, і з нього почнеться відображення символів, набраних на клавіатурі. 

*Вивести значення змінної* на екран можна за допомогою інструкції вигляду **cout << ім'я-змінної;**

**Приклад**. Після введення значень трьох дійсних змінних за інструкцією **cin >> a >> b >> c;** можна вивести їх на екран, відокремивши пробілами:

**cout << a; cout << ' '; cout << b; cout << ' '; cout << c;**

Ті самі дії можна задати й однією інструкцією: **cout << a << ' ' << b << ' ' << c;**

Якщо введено 1, –3, 2, то буде надруковано **1 –3 2.** 

**Вправа 2.12**. Що буде виведено на екран, якщо під час виконання програми введено символи **15 n** та натиснуто на клавішу

**Enter**?

**#include <iostream> using namespace std; int main() { int n; char a; cin >> n >> a;**

**cout<<"n="<<n<<' '<<"a="<<a<<endl; system("pause"); return 0;**

**}**

#### 2.6.3. Присвоювання: операція, інструкція, вираз

Надати значення змінній можна за допомогою **операції** **присвоювання значення змінній**, яка копіює значення деякого виразу в змінну. Знаком операції є **=**, її можна задати в **інструкції присвоювання** (*assignment statement*): ***ім'я-змінної = вираз;***

Спочатку обчислюється вираз праворуч, потім його значення записується в змінну, указану ліворуч. Найпростішими виразами є константи та імена змінних, складніші описано нижче.

##### Приклади

1. Якщо є символьна змінна **char c;**, то інструкція **c='0';** надає їй значення **'0'**.
2. Якщо є ціла змінна **int i;**, то перша з інструкцій **i=13; cout<<i;** надає їй значення 13, а друга виводить символи **13** на екран.
3. Нехай дійсні змінні **double a, b, c;** зображують квадратне рівняння *ax*2+*bx*+*c* 0. Щоб задати рівняння *x*2-3*x*+2 0, присвоїмо їм значення відповідно 1, -3, 2, а потім виведемо на екран.

**a=1.0; b=-3.0; c=2.0; cout << a << ' ' << b << ' ' << c;**

Під час виконання має бути надруковано **1 –3 2.** 

В інструкції присвоювання ***ім'я-змінної = вираз*;**ліворуч ім'я змінної позначає ділянку пам'яті, в яку значення записується, а у виразі праворуч – значення, що добувається з ділянки.

Добування значення змінної називається її **розіменуванням**.

##### Приклади

1. Якщо змінні **a** та **b** однотипні, то інструкція **a = b;** копіює значення змінної **b** у змінну **a**.
2. Щоб обміняти місцями значення двох однотипних змінних **a** та **b**, скористаємося третьою змінною **c** того самого типу. Припустимо, що змінні **a** та **b** мають початкові значення *v*1 і *v*2, і праворуч укажемо значення, яких набувають змінні після виконання інструкцій. Початкове значення змінної **c** вважається невизначеним і позначене знаком **?**. c

a

B

*v*

1

*v*

2

*v*

1

*v*

2

*v*

2

*v*

2

*v*

2

*v*

1

Спочатку ? **c=a;** *v*1 **a=b;** *v*1 **b=c;** *v*1



Перед тим, як використовувати значення змінної, необхідно забезпечити, щоб раніше ця змінна його отримала (під час присвоювання або введення), інакше можливі непередбачувані наслідки. Зазвичай компілятори лише попереджають, що змінна не отримала значення перед використанням, тому за цим має стежити програміст.

**Вправа 2.13**. Написати інструкції оголошення дійсних змінних **a, b, c,** присвоювання їм початкових значень та обміну місцями цих значень за колом: значення **a** пересувається до **c**, **c** до **b**, **b** – до **a.** *Порада*: почніть із проектування коду.

Запис вигляду *ім'я-змінної = вираз* (без ; в кінці) називається **виразом присвоювання**. Його значенням є значення, присвоєне змінній.

Вираз присвоювання можна записати праворуч від знака **=**, наприклад, **a = b = 2**. Для наочності "внутрішнє" присвоювання можна записати в дужках: **a = (b = 2)** – дії будуть ті самі. Інструкція **a = b = 2;** задає в програмі ті самі дії, що й послідовність **b = 2; a = b;.** Довжину ланцюгів присвоювань не обмежено.

Інструкції та вирази присвоювання можуть бути частиною інструкцій інших різновидів (наприклад, інструкцій керування), описаних нижче.

Сукупність змінних, імена яких означено в програмі, називається **пам'яттю програми**[[1]](#footnote-1), а відповідність між іменами змінних та їх станами – **станом пам'яті програми**. Присвоювання значення змінній **змінює стан пам'яті програми**.

Ім'я змінної є прикладом загальнішого поняття – ***l*-вираз**. Так називають вираз, який можна записати ліворуч (*left*) у виразі присвоювання. Він позначає змінну. Вираз, що позначає значення й записується праворуч (*right*) у виразах присвоювання, називається ***r*-виразом**.

###### 2.6.4. Ініціалізація змінних

Присвоїти початкове значення змінній можна в її означенні (**ініціалізувати змінну**), дописавши до імені змінної знак **=** і вираз. У виразі можуть бути константи та імена змінних (звичайно, якщо вони вже отримали значення). **Приклади**

1. За інструкцією **int a = 44, b = a, c;** створені змінні **a** та **b** отримують значення **44**, а значення змінної **c** залишається невизначеним, тому використовувати її значення можна тільки після того, як вона його отримає.
2. Інструкція **int i=j=k=0;** створює змінні **i**, **j**, **k**, які відразу отримують значення **0**. 

1. Далі побачимо, що в процесі виконання програми змінні можуть додаватися до її пам'яті або, навпаки, вилучатися. [↑](#footnote-ref-1)