**Реферат на тему:**

**Оператор присвоювання**

В будь-якій мові програмування можна виділити чотири типи елементів, що використовуються при побудові описів програм:

* символи,
* слова,
* вирази,
* **команди (оператори).**

Команда - це вказівка про виконання деякої дії. При написанні програм команди називають **операторами**, а величини, що використані в команді - операндами.

**Оператор присвоювання** має вигляд:

*ім'я* *змінної* := *вираз*

(знак присвоювання ":=" – це лексема, яку не слід плутати зі знаком порівняння "="). Оператор присвоювання позначає:

1) обчислити значення виразу, записаного праворуч;

2) записати це значення в змінну, позначену ім'ям.

Отже, описані дії є *семантикою оператора присвоювання*.

**Приклад**. Якщо ім'я z означено як **var** z : integer, то оператор присвоювання z:=11\*(10+1) позначає обчислення значення 121 і запис його в змінну з ім'ям z. Після його виконання змінна з ім'ям z має значення 121.

Пара вигляду (ім'я, значення) називається ***станом* *змінної***, наприклад, (z,121) – стан змінної z після виконання оператора присвоювання. Таким чином, *після присвоювання змінній нового значення її стан змінюється*.

Оператори присвоювання (і не тільки вони) в програмі записуються один за одним і відокремлюються роздільником ";", наприклад, z:=1; t:=2 (звичайно, за умови, що z і t означено як імена змінних типу integer). Оператори, записані один за одним, утворюють ***послідовність* *операторів***. Кожен з них задає зміну стану хоча б однієї зі змінних.

Сукупність змінних, чиї імена означено в програмі, називається ***пам'яттю* *програми***. Сукупність станів змінних називається ***станом* *пам'яті* *програми***. Зміна стану однієї зі змінних змінює стан пам'яті програми, тому оператор присвоювання задає *зміну стану пам'яті програми*. Далі ми розглянемо інші види операторів, але

*всі оператори задають зміну станів пам'яті програми*. *Ця зміна і є їхньою семантикою.*

Виконання операторів програми можна ***проімітувати***, указавши їх послідовність і послідовність станів пам'яті програми, що утворюються в результаті їх виконання. Якщо в процесі виконання програми змінна ще не одержала значення, то воно вважається невизначеним і позначається "?". Наприклад, ось програма та подання її імітації:

|  |  |  |  |
| --- | --- | --- | --- |
| **виконуваний оператор** | **стан пам'яті** | | |
|  | **x** | **y** | **z** |
|  | ? | ? | ? |
| z := 1 | ? | ? | 1 |
| x := 3 | 3 | ? | 1 |
| y := 15 | 3 | 15 | 1 |
| x := 10 | 10 | 15 | 1 |

**program** a2(input, output);

**var** x, y, z : integer;

**begin**

z := 1;

x := 3;

y := 15;

x := 10

**end**.

Звернімо увагу на те, що нові значення змінних записуються замість старих.

Арифметичний оператор присвоювання на Паскалі має наступний формат:

<числова перемінна>:=<арифметичне вираз>

Арифметичне вираз може містити числові константи і перемінні, знаки арифметичних операцій, круглі дужки. Крім того, в арифметичних виразах можуть бути присутнім функції.   
Знаки основних арифметичних операцій записуються так:

+ додавання,

- віднімання,

\* множення,

/ ділення.

Запис арифметичних виразів на Паскалі схожа на звичайний математичний запис. На відміну від математики, де часто пропускається знак множення (наприклад, пишуть 2a), у Паскалі цей знак пишеться обов'язково: 2\*a.   
Наприклад , математичний вираз   
a 2 + b2 - 12c   
На Паскалі пишеться так:   
A\*A+B\*B-12\*C.   
Цей вираз можна записати інакше:   
SQR(A) + SQR(B) - 12\*C.   
Тут використана функція зведення в квадрат - SQR . Аргументи функцій завжди пишуться в круглих дужках.   
Послідовність виконання операцій визначається по їхніх пріоритетах (старшості). До старших операцій відносяться множення (\*), ділення (/). Операції додавання і вирахування - молодші. У першу чергу виконуються старші операції. Кілька операцій однакового старшинства, записані підряд, виконуються в порядку їхнього запису ліворуч праворуч. Приведений вище арифметичне вираз буде обчислюватися в наступному порядку (порядок обчислень зазначений цифрами зверху):

1 4 2 5 3

A\*A + B\*B - 12\*C.

Круглі дужки в арифметичних виразах впливають на порядок виконання операцій. Як і в математиці, у першу чергу виконуються операції в дужках. Якщо є в наявності декілька пар вкладених дужок, то спочатку виконуються операції в самих внутрішніх дужках. Дужки розкриваються зсередини назовні. Наприклад :

6 1 3 2 4 5

A + (( C - D ) / ( 2 + K ) - 1) \* B .

## Необхідно суворе дотримання правопису (синтаксису) програми. Зокрема, у Паскалі однозначно визначене призначення знаків пунктуації. Крапка з комою (;) ставиться наприкінці заголовка програми, наприкінці розділу опису перемінних, є роздільником операторів. Перед словом end крапку з комою можна не ставити. Кома (,) є роздільником елементів у всіляких списках: список перемінних у розділі опису, список що вводяться і виведених величин. Суворий синтаксис у мові програмування необхідний тому, що *комп'ютер є формальним виконавцем програми* . Якщо, допустимо, роздільником у списку перемінних повинна бути кома, то будь-який інший знак буде сприйматися як помилка. Якщо крапка з комою є роздільником операторів, то як оператора комп'ютер сприймає всю частину тексту програми від однієї крапки з комою до іншої. Якщо програміст забув поставити : <;> між якимись двома операторами, то комп'ютер буде приймати їх за один з неминучою помилкою. У програму на Паскалі можна вставити коментарі. Коментар - це пояснення до програми, що записується у фігурних дужках. У коментарях можна використовувати російські букви. На виконання програми коментар ніяк не впливає. Помітимо, що в Паскалі немає розходження між рядковими і прописними буквами. Різниця тільки зовнішня. Наприклад, для Паскаля тотожні наступні варіанти запису слова: begin, Begin, BEGIN, BeGi. Використання рядкових чи прописних букв - справа смаку програміста.

## Запис арифметичних виразів

Виразами називаються алгоритмічні конструкції мови, що визначають правила для обчислення значень змінних величин. Вираз складається з операндів: змінних, констант, назв функцій, що відокремлюються один від одного за допомогою операторів.

У наступних таблицях наведені типи операндів і результати для бінарних і унарних арифметичних операцій, а також основні арифметичні функції.

Будь-який операнд, тип якого є піддіапазоном порядкового типу, опрацьовується так, якби він був порядкового типу.

Якщо обидва операнди в операторах +, -, \*, **div** або **mod** є операндами цілого типу, то тип результату є таким же, як спільний тип обох операндів.

Якщо при використанні оператора збереження знака або оператора заперечення знака операнд має цілий тип, то результат є того ж цілого типу. Якщо операнд дійсного типу, то тип результату є **Real**.

Значення виразу ***x/y*** завжди є **Real** незалежно від типів операндів. Якщо ***y***дорівнює 0, то виникає помилка.

Значення виразу **i div j** являє собою цілу частину від ділення ***i/j***. Якщо ***j*** дорівнює 0, то виникає помилка. Наприклад, 25 **div** 4 = 6, 21 **div** 3 = 7.

Оператор **mod** повертає залишок, отриманий при діленні двох його операндів. Наприклад, 25 **mod** 4 = 1, 21 **mod** 3 = 0.

Знак результату оператора **mod** є тим же, що і знак змінної **i**. Якщо **j** дорівнює 0, то виникає помилка.

**Таблиця 1 Бінарні арифметичні операції**

|  |  |  |  |
| --- | --- | --- | --- |
| Оператор | Операція | Тип  операндів | Тип  результату |
| + | додавання | цілий | цілий |
| дійсний | дійсний |
| – | віднімання | цілий | цілий |
| дійсний | дійсний |
| \* | множення | цілий | цілий |
| дійсний | дійсний |
| / | ділення | цілий | дійсний |
| дійсний | дійсний |
| div | цілочисельне ділення | цілий | цілий |
| mod | залишок цілочисельного ділення | цілий | цілий |

**Таблиця 2. Унарні арифметичні операції**

|  |  |  |  |
| --- | --- | --- | --- |
| Оператор | Операція | Тип операндів | Тип результату |
| + | збереження знака | цілий | цілий |
|  |  | дійсний | дійсний |
| - | заперечення знака | цілий | цілий |
|  |  | дійсний | дійсний |

**Таблиця 3. Основні арифметичні функції.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Математичний запис | Запис в  ПАСКАЛІ | Тип  операнду | Тип результату |
| 1. | sin x | sin(x) | real | real |
| 2. | cos x | cos(x) | real | real |
| 3. | ex | exp(x) | real | real |
| 4. | ln x | ln(x) | real | real |

|  |  |
| --- | --- |
| Математичний запис | Запис мовою ПАСКАЛЬ |
|  | (a+b)/(c\*d) |
|  | exp(x+1)\*(a-x) |
|  | sqr(sin(x\*x\*x)) |
|  | sin((2\*x)/(x+1))/cos(((2\*x)/(x+1)) |
|  | exp(1/3\*ln(x+1)) |

Зауважимо, що можливі й інші вірні варіанти запису виразів.