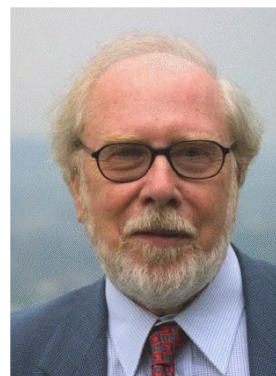


Моделювання логічних операцій на мові програмування Pascal.

Алгоритмічна мова Pascal розроблена професором Цюріхського технологічного інституту Ніклаусом Віртом у 71 роках для навчання студентів структурному програмуванню. Ідеї, закладені в основу мови, дозволили Borland International значно розширити алгоритмічні засоби зручний інтерфейс (меню команд) і висока швидкість компіляції (процесу перекладу тексту програми в машинні зробили мову Turbo Pascal одною з самих поширених серед початківців і професіональних програмістів.



1969-
фірмі
мови, а
коди)

Існує ряд об'єктивних причин, які обумовили видатний успіх мови Pascal. Серед них у першу чергу потрібно вказати такі:

- Мова в природній і елегантній відбила найважливіші сучасні концепції технології розробки програм.

- Завдяки своїй компактності, концептуальній цілісності й ортогональності понять, а також вдалому оригінальному опису, запропонованому автором мови, Pascal виявився дуже легким для вивчення й освоєння.



формі

- Незважаючи на відносну прототипність мови, вона виявилась придатною для дуже широкого спектру застосунків, у тому числі для розробки дуже великих і складних програм, наприклад, операційних систем.

- Pascal дуже технологічний для реалізації практично усіх, у тому числі і нетрадиційних, машинних архітектур. Стверджується, що розробка Pascal-транслятора «майже» не перевищує за трудомісткістю гарної дипломної роботи випускника ВНЗу.

- Мова Pascal стандартизована в багатьох країнах, а у 1983 році було прийнято міжнародний стандарт (ISO 7185:1983).

Математична логіка є одною із найголовніших математичних апаратів програмування. У мові Pascal визначений логічний тип даних – Boolean і введені логічні операції, за допомогою яких можна створювати складні логічні висловлювання.

Змінні логічного типу описуються в розділі опису змінних. Для зберігання логічної змінної відводиться один байт пам'яті.

`var a, b, c : Boolean;`

Операції, визначенні над логічними змінними

1. Операція присвоєння

`t := false; p := t; q := 5 < x;`

2. Операція виведення

`write (t); writeln (q); write (5 > 2)`

Величини логічного типу можна присвоювати, виводити, але не можна вводити

процедурою read.

3. Логічні (булевські) операції

Операція	Пояснення
<i>and</i>	Логічне множення (кон'юнкція)
<i>or</i>	Логічне додавання (диз'юнкція)
<i>not</i>	Логічне заперечення (не)
<i>xor</i>	Логічне ділення (виключаюче або)

Заперечення. Якщо логічна величина є запереченням логічного висловлювання А, то С істинне, якщо А хибне, і хибне, якщо А істинне.

Логічне множення. Якщо А і В істинні, то С також істинне. Якщо ж хоча б одне з них хибне, то С також хибне.

Логічне додавання. Якщо А і хибні, то С також хибне. Якщо ж хоча б одне із логічних висловлювань А і В істинне, то С також істинне.

Логічне ділення (або виключаючи або). Ця логічна операція встановлює відповідність між логічними висловлювання А і В та логічною величиною наступним чином: С хибне, якщо А і В одночасно істинні, або одночасно хибні.

Таблиця істинності для всіх логічних операцій

A	B	not A	A and B	A or B	A xor B
true	true	false	true	true	false
true	false	false	false	true	true
false	true	true	false	true	true
false	false	true	false	false	false

Приклади логічних висловлювань:

(a and b) or (b and not c)

(a xor b) and (a or c) xor (not h)

Вміння будувати логічні вирази (логічні формули), обчислювати їх значення, виконувати над ними тотожні перетворення потрібні при вивченні різних тем інформатики: при побудові алгоритмів, в програмуванні, при вирішенні логічних завдань, конструюванні запиту при роботі з базами даних, при роботі з електронними таблицями та т.п.

4. Стандартні булевські функції

- *odd* (x) = true, якщо x – непарне (x цілий тип);
- *eoln* (x) = true, якщо зустрівся кінець рядка текстового файлу x;
- *eof* (x) = true, якщо зустрівся кінець файлу x.

В інших випадках ці функції приймають значення false.

• *pred*(x) — функція, що визначає попереднє значення для даного типа. Наприклад, Pred(5)=4, Pred(True)=False, Pred('B')='A'.

• *Succ*(x) — функція, що визначає наступне значення для даного типа. Наприклад, Succ(5)=6, Succ(True)=False, Succ('A')='B'.

- *Odd*(x) — число перевіряється на непарність. Результат дорівнює true, якщо

аргумент непарний, і false — в протилежному випадку.

Логічне висловлювання – це таке висловлювання, яке може приймати тільки два значення: істинне (true) і хибне (false). На мові Pascal логічними висловлюваннями можуть бути:

- спеціальні логічні змінні (тип Boolean);
- висловлювання, які містять порівняння (наприклад, $a < b$);
- складні висловлювання, що містять висловлювання двох описаних раніше типів, які з'єднуються знаками логічних операцій та дужками.

Спочатку виконуються логічні оператори (*and*, *or*, *xor*, *not*), а вже потім оператори відношень ($>$, $>=$, $<$, $<=$, $<>$, $=$), тому потрібно не забувати розставляти дужки у складних логічних висловлюваннях.

Приклади складних логічних висловлювань:

$(a < b) \text{ and } (f \text{ or } (h < 5))$

$f \text{ xor } (5 = 6 * i)$

$(s < 6) \text{ or } (q > 8)$

Уміння будувати складні висловлювання потрібне при конструюванні запиту пошуку в Інтернеті, при побудові алгоритмів і написанні програм на будь-якій алгоритмічній мові. Більше того, це вміння можна віднести до загальношкільних вмінь, тому що воно пов'язане з побудовою складних умовиводів (міркувань, отримань висновків). В основі цього вміння лежать знання основних логічних операцій і вміння визначати істинність складних висловлювань.

У мові Pascal для розв'язування задач спочатку слід виразити операції імплікації та еквівалентності через операції *or* і *and*. Операція слідування $A \rightarrow B$ може бути записати наступним чином: $\text{not } a \text{ or } b$. А операція еквівалентності $A \leftrightarrow B$ так: $(a \text{ or } (\text{not } b)) \text{ and } ((\text{not } a) \text{ or } b)$ або $(a \text{ and } b) \text{ or } ((\text{not } a) \text{ and } (\text{not } b))$.

Крім типу Boolean в Pascal введені ще три типи булевих типів – **bytebool** (займає 1 байт), **wordbool** (займає 2 байти) і **longbool** (займає 4 байти).

Для всіх булевих типів значенню **false** відповідає 0, а значення **true** – будь-яке ненульове значення. Логічні змінні, які належать різним булевим типам, ведуть себе по-різному при виконанні над ними операцій.

1. Приклади запису логічних висловлювань, істинних при виконанні вказаних умов.

Умова	Запис на мові Pascal
Дробова частина дійсного числа дорівнює нулю	$\text{int}(a) = 0$
Ціле число a — парне	$\text{mod}(a, 2) = 0$
Ціле число a — непарне	$\text{mod}(a, 2) = 1$
Ціле число k кратне семи	$\text{mod}(a, 7) = 0$
Кожне із чисел a , b додатне	$(a > 0) \text{ and } (b > 0)$
Тільки одне із чисел a , b додатне	$((a > 0) \text{ and } (b \leq 0)) \text{ or } ((a \leq 0) \text{ and } (b > 0))$
Хоча б одне із чисел a , b , c є від'ємним	$(a < 0) \text{ or } (b < 0) \text{ or } (c < 0)$
Число x задовільняє умову $a < x < b$	$(x > a) \text{ and } (x < b)$

Число x належить проміжку $[1, 3]$	$(x \geq 1) \text{ and } (x \leq 3)$
Цілі числа a і b мають однакову парність	$((\text{mod}(a, 2)=0) \text{ and } (\text{mod}(b, 2)=0) \text{ or } ((\text{mod}(a, 2)=1) \text{ and } (\text{mod}(b, 2)=1)))$
Точка з координатами (x, y) належить колу з радіусом r та центром (a, b)	$(x-a)^2 + (y-b)^2 < r^2$
Рівняння $ax^2 + bx + c = 0$ не має дійсних коренів	$b^2 - 4ac < 0$
Точка (x, y) належить першій або третій чверті	$((x > 0) \text{ and } (y > 0)) \text{ or } ((x < 0) \text{ and } (y > 0))$
Цілі числа a і b є взаємно протилежними	$a = -b$
Цілі числа a і b є взаємно оберненими	$a * b = 1$
Число a більше середнього арифметичного чисел b, c, d	$a > (b+c+d) / 3$
Число a не менше середнього геометричного чисел b, c, d	$a \geq \exp(\ln(b*c*d)/3)$
Хоча б одна із логічних змінних $F1$ і $F2$ має значення <i>так</i>	$F1 \text{ or } F2$
Обидві логічні змінні $F1$ і $F2$ мають значення <i>так</i>	$F1 \text{ and } F2$
Обидві логічні змінні $F1$ і $F2$ мають значення <i>ні</i>	$\text{not } (F1 \text{ and } F2)$
Логічна змінна $F1$ має значення <i>так</i> , а логічна змінна $F2$ має значення <i>ні</i>	$F1 \text{ and not } F2$
Тільки одна із логічних змінних $F1$ і $F2$ має значення <i>так</i>	$(F1 \text{ and not } F2) \text{ or } (F2 \text{ and not } F1)$

2. Приклад поведінки логічних змінних, у коментарях відображається результат.

```

program logic1;
var a : Boolean;
    b : wordbool;
begin
a:= false;
a:= pred(a);
writeln (a, ' ', ord(a));    {true 255}
writeln (a=true);           {true}
b:= false;
b:= pred(b);
writeln (b, ' ', ord(b));    {true -1}
a:= true;
a:= succ (a);
writeln (a, ' ', ord(a));    {true 2}
b:= true;
b:= succ (b);
writeln (b, ' ', ord(b));    {false 0}
readln;
end.
```

The screenshot shows the Turbo Pascal IDE. The top window, titled 'NONAME00.PAS', contains the following code:

```

program logic1;
var a: boolean; b: wordbool;
begin
a:=false;
a:=pred(a);
writeln(a, ' ', ord(a));
writeln (a = true);
b:=false;
b:=pred(b);
writeln(b, ' ', ord(b));
a:=true;
a:=succ(a);
writeln(a, ' ', ord(a));
b:=true;
b:=succ(b);
writeln(b, ' ', ord(b));
readln;
end.

```

The bottom window shows the output of the program:

```

Turbo Pascal Version 7.0 Copyright (c) 1983,92 Borland International
TRUE -1
FALSE -1
TRUE 1
TRUE 2
TRUE 1

```

3. Скласти таблицю істинності логічного висловлювання $(A \vee B) \wedge \bar{A}$. Розв'язування

На мові програмування Pascal дане висловлювання для логічних змінних a, b буде мати наступний вигляд: (a or b) and not a. Змінній c будемо присвоювати значення висловлювання для різних значень операндів a і b.

Програма:

```

program logic2;
var a, b, c : Boolean;
begin
for a:= false to true do
  for b:= false to true do
    begin
c:= (a or b) and not a;
writeln (a, ' ', b, ' ', c);
end;
readln
end.

```

Запустивши програму на виконання, матимемо наступний результат:

```

FALSE FALSE FALSE
FALSE TRUE TRUE
TRUE FALSE FALSE
TRUE TRUE FALSE

```

У цій таблиці перший стовпчик відповідає значенням змінної a, а другий і третій відповідно змінним b, c.

Завдання для самостійного розв'язування

Запишіть логічні висловлювання, істинні тільки при виконанні вказаних умов:

- 1) x належить відрізку $[a, b]$;
- 2) x не лежить відрізку $[a, b]$;
- 3) x належить відрізку $[a, b]$ або відрізку $[c, d]$;
- 4) x не належить відрізкам $[a, b]$ і $[c, d]$;
- 5) ціле число k є непарним числом;
- 6) ціле k є трьохзначним числом, яке кратне п'яти;
- 7) елемент a_{ij} двовимірного масиву знаходиться на перетині непарного рядка і парного стовпця;
- 8) прямі $a_1x + b_1y + c_1 = 0$ і $a_2x + b_2y + c_2 = 0$ паралельні;
- 9) із чисел a, b, c меншим є c , а більшим b ;
- 10) серед чисел a, b, c, d є взаємно протилежні;
- 11) серед цілих чисел a, b, c є хоча б двоє парних;
- 12) з відрізків довжиною a, b, c можна побудувати трикутник;
- 13) трикутники зі сторонами a_1, b_1, c_1 і a_2, b_2, c_2 подібні;
- 14) точка з координатами (x, y) належить внутрішній області трикутника з вершинами $A(0,5)$, $B(5,0)$ і $C(1,0)$;
- 15) чотирикутник зі сторонами a, b, c і d є ромбом.