

OPERATORY ARYTMETYCZNE

Podział operatorów w Pascal:

- jednoargumentowe i dwuargumentowe
- oraz na arytmetyczne, bitowe, logiczne, mnogościowe, relacji;
- jest również operator łańcuchowy, wskazania, konwersji.

Operatory arytmetyczne umożliwiają wykonywanie obliczeń.

Operator jednoargumentowy jest stosowany do zmiany znaku wartości, co zilustrowano w listingu:

```
var x :integer;
begin
    X:=10;    {operator podstawienia dla zmiennej x została podstawiona wartość 10}
    writeln ('1:wartosc zmiennej x wynosi: ',x);
    x:= -x;
    writeln ('2: wartosc zmiennej x wynosi : ',x);
    x:= -x;
    writeln ('3: wartosc zmiennej x wynosi: ',x);
    readln;
end.
```

Za każdym razem wartość zmiennej x jest przeciwna.

Arytmetyczne operatory dwuargumentowe: dodawania, odejmowania, mnożenia i dzielenia.

Lista operatorów arytmetycznych

Operator	Ilość argumentów	Przykład użycia	Znaczenie
-	jednoargumentowy	-a	zmiana znaku
+	jednoargumentowy	+x	zachowanie znaku
-	dwuargumentowy	a-b	różnica a,b
+	dwuargumentowy	a+b	suma a ,b
*	dwuargumentowy	a*b	iloczyn a,b
/	dwuargumentowy	a/b	iloraz a,b
div	dwuargumentowy	a div b	dzielenie całkowite
mod	dwuargumentowy	a mod b	reszta z dzielenia

Operatory można również podzielić na **addytywne** (dodawania i odejmowania, w tym or i xor są podstawowymi spośród operatorów arytmetycznych) oraz **multiplikatywne** (mnożenia, dzielenia, w tym div, mod, and, shl, shr) o priorytecie wyższym od operatorów addytywnych.

OPERATORY LOGICZNE

PRZYDATNE DO wykonanie operacji logicznych na wartościach typu **Boolean** oraz wartościach całkowitych:

Operator	Operacja	Przykład
not (jednoargumentowy)	negacja	not prawda=falsz
and	konjunkcja	prawda and falsz=falsz
or	alternatywa	prawda or falsz=prawda
xor	różnica symetryczna	prawda xor prawda=falsz
shl	przesunięcie w lewo	7 shl 1 = 14
shr	przesunięcie w prawo	7 shr 2 = 1

PRZYKŁAD DZIAŁANIA OPERATORÓW LOGICZNYCH:

```
var
    x,y :integer;
begin
```

```

Writeln('Podaj dla x wartosc 1 albo 2');
readln(x);
Writeln('Podaj dla y wartosc 1 albo 2');
readln(y);
if (x=1) and (y=1) then
    Writeln ('zachodzi konjunkcja')
else
    writeln('Dla 'x,'=1 and 'y,'=1 konjunkcja nie zachodzi');
if (x=1) or (y=1) then
    Writeln ('zachodzi alternatywa')
else
    writeln('Dla 'x,'=1 or 'y,'=1 alternatywa nie zachodzi');
if (x=1) xor (y=1) then
    writeln ('zachodzi roznica symetryczna')
else
    writeln('Dla 'x,'=1 xor 'y,'=1 roznica symetryczna nie zachodzi');
readln;
end.

```

DZIAŁANIE OPERATORÓW LOGICZNYCH **shl** i **shr** - dotyczą przesunięcia bitowego. Jeżeli mamy wartość dziesiętną 7 przedstawioną binarnie 0111, to przesunięcie bitowe w lewo shl spowoduje uzyskanie wartości 14 przedstawia tabela:

Przesunięcie w lewo o 1					
shl	y=1	<-----			
	x=7	0	1	1	1
Dziesiętnie = 14		1	1	1	0

Przesunięcie w prawo o 2					
shr	y=2	----->			
	x=7	0	1	1	1
Dziesiętnie=1		0	0	0	1

Do wykonywania operacji bitowych służą również operatory and, or, xor.

PRZYKŁAD DZIAŁANIA:

Koniunkcja (X AND Y)					
And („i”)	y=7	0	1	1	1
	x=4	0	1	0	0
Dziesiętnie = 4		0	1	0	0

Alternatywa (X OR Y)					
OR („lub”)	y=7	0	1	1	1
	x=4	0	1	0	0
Dziesiętnie = 7		0	1	1	1

Alternatywa wykluczająca (X XOR Y)					
XOR	y=7	0	1	1	1
	x=4	0	1	0	0
Dziesiętnie = 3		0	0	1	1

OPERATORY RELACYJNE

Operatory relacyjne to operatory dwuargumentowe – służą do badania relacji zachodzących pomiędzy operandem po lewej stronie a operandem znajdującym się po prawej stronie operatora

LISTA OPERATORÓW:

Operator	Funkcja operatora	Przykład
<	Mniejszy	X<Y
<=	Mniejszy lub równy	X<=Y
>	Większy	X>Y
>=	Większy lub równy	X>=Y
<>	Różny	X<>Y
=	równy	X=Y

Operatory relacyjne mogą służyć do porównywania zarówno liczb (typy całkowite i rzeczywiste) jak również typu logicznego, typu znakowego czy łańcuchowego. NP:

'a' = 'A' lub 'imie' = 'imie'

If 'imie' = 'imie' then writeln('rowne');

If 'IMIE' <> 'imie' then writeln('nierowne');

UWAGA: Efektem zastosowania operatora relacyjnego zawsze jest wartość logiczna. Jeśli porównać dwie wartości to zawsze wynikiem takiego porównania jest wartość TRUE lub FALSE w zależności czy porównanie było prawdziwe:

Np:

```
Var wartosc_logiczna:boolean;  
begin  
    wartosc_logiczna := 1 = 2; (*W rezultacie da wartość FALSE*)  
    wartosc_logiczna := 1 <> 2; (*W rezultacie da wartość TRUE*)  
End.
```