

Bazele programării I

Date. Tipuri de date elementare

Date. Manipularea datelor

Prin **dată** se înțelege orice entitate asupra căreia poate opera calculatorul.

La executarea programului datele se supun unor modificări.

Manipularea datelor:

- operatori de citire a datelor;
- operatori de prelucrare a datelor;
- operatori de afișare a datelor.

Variabile. Definiții

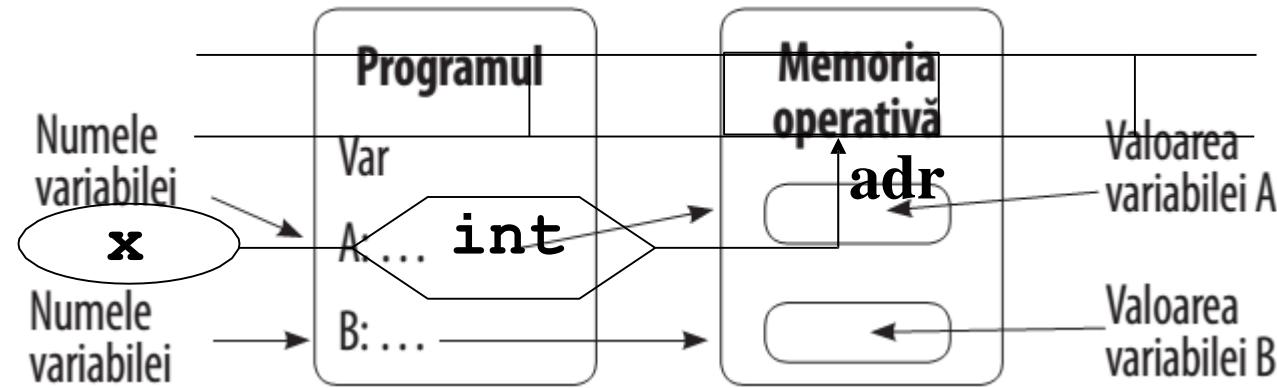
Prin **variabilă** se înțelege o informație (o dată) a cărei valoare se poate modifica pe parcursul execuției unui program.

Variabila este o locație de memorie care poate păstra o valoare dintr-o mulțime prestabilită de valori.

Domeniul de definiție al variabilei - mulțimea prestabilită de valori.

Locația - zonă de memorie pentru păstrarea oricărei valori din domeniul de definiție.

Locație abstractă a memoriei - set de locații fizice a memoriei operative a calculatorului.



Atributele variabilei

Atribut - însușire (esențială) a unui obiect (fără de care acesta nu poate exista și nici nu poate fi conceput).

- Nume;
- Tip de date;
- Valoare;
- Durată de existență (timp de viață);
- Domeniu de vizibilitate;
- Adresă.

Numele și tipul de date a variabilei

Numele variabilei este unul din atributele ei principale și reprezintă un sir de caractere.

Tipul de date a variabilei definește domeniul de valori posibile a variabilei, numit *domeniul de definiție* a variabilei și *multimea operațiilor* ce pot fi efectuate cu aceste valori.

Un tip de date precizează:

- multimea valorilor pe care le poate lua o dată;
- operațiile care se pot efectua asupra ei.

Atributele variabilei

Valoarea (conținutul) variabilei - conținutul locației abstracte de memorie care reprezintă variabila.

Durata de existență a variabilei - intervalul de timp pe parcursul căruia variabila este legată de locația memoriei.

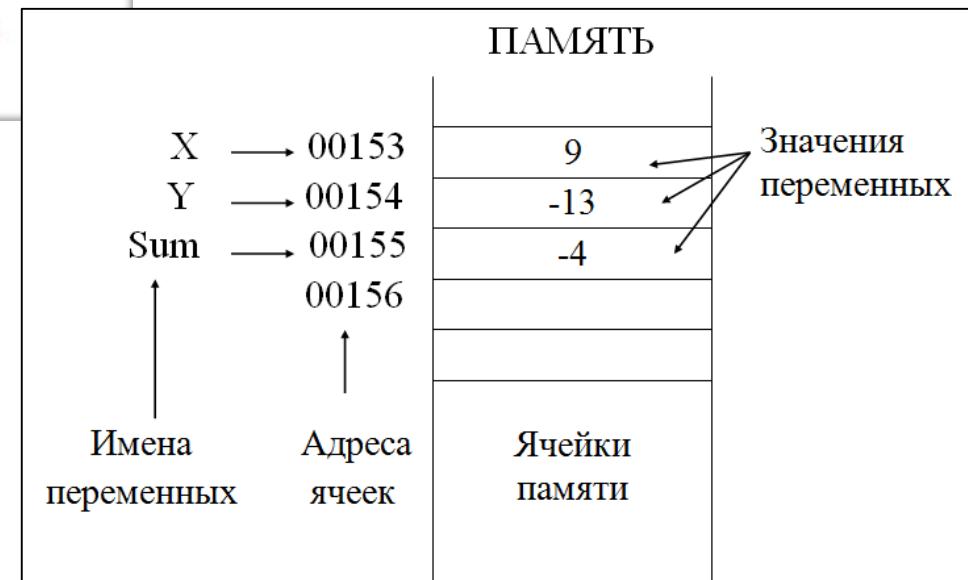
Domeniul de vizibilitate a variabilei reprezintă totalitatea instrucțiunilor programului în care variabila poate fi folosită.

Variabile globale – accesibile din orice loc a programului, *Variabile locale* - numai în cadrul unor componente ale programului.

Adresa variabilei – numărul celulei din memoria operativă unde se rezervă un anumit spațiu pentru păstrarea valorii variabilei.

Adresa variabilei

Computer		Programmers		
Address	Content	Name	Type	Value
90000000	00			
90000001	00			
90000002	00			
90000003	FF			
90000004	FF	sum	int (4 bytes)	000000FF (255 ₁₀)
90000005	FF			
90000006	1F	age	short (2 bytes)	FFFF (-1 ₁₀)
90000007	FF			
90000008	FF			
90000009	FF			
9000000A	FF			
9000000B	FF			
9000000C	FF			
9000000D	FF			
9000000E	90			
9000000F	00			
90000010	00	ptrSum	int* (4 bytes)	90000000
90000011	00			



Prelucrarea variabilelor

1. Crearea (declararea) variabilei;
2. Inițierea variabilei (atribuirea valorii variabilei);
3. Utilizarea variabilei în algoritm.

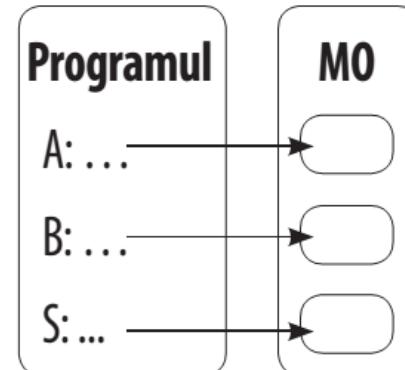
Declararea variabilei

numele variabilei: tipul de date a variabilei

La **declararea variabilei** se indică:

- numele variabilei (să fie sugestiv, să indice destinația ei);
- tipul de date (se indică cu ajutorul unui nume, de exemplu, Integer).

Exemple: A: Integer
 B: Integer
 S: Integer



Inițierea variabilei

Atribuirea valorii unei variabile poate fi realizată prin:

- operații de atribuire;
- operații de citire.

Inițierea variabilei

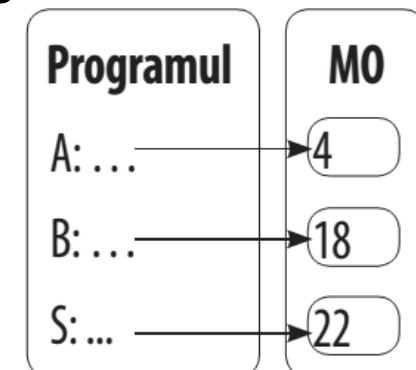
$<\text{nume_variabilă}> := <\text{expresie}>$

La executarea operației de atribuire:

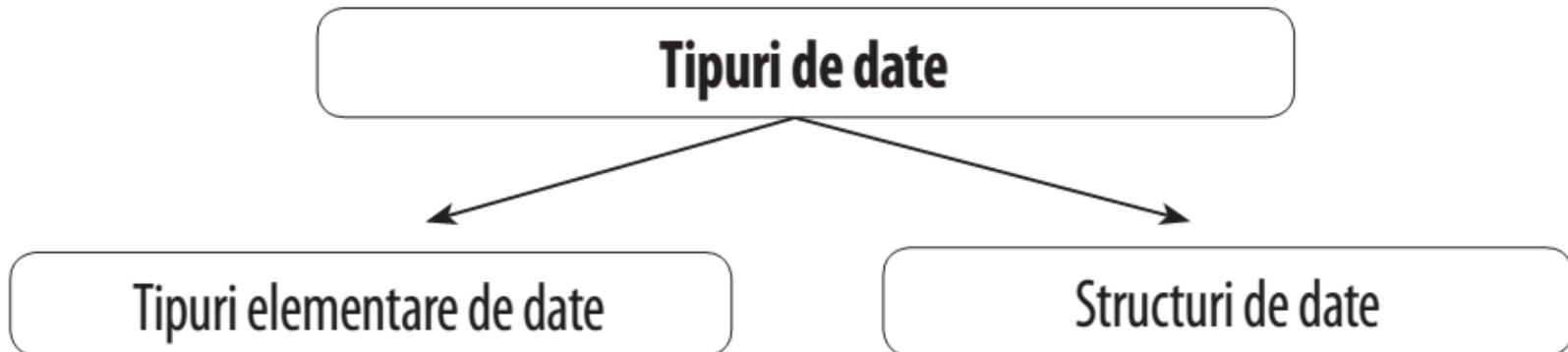
- se calculează valoarea expresiei;
- valoarea calculată se atribuie variabilei, dacă ea aparține multșimii de valori a tipului de date. În caz contrar, se va semnala eroare de tip run-time.

Exemple:

1. A:= 4
2. B:=18
3. S:=A+B



Tipuri de date

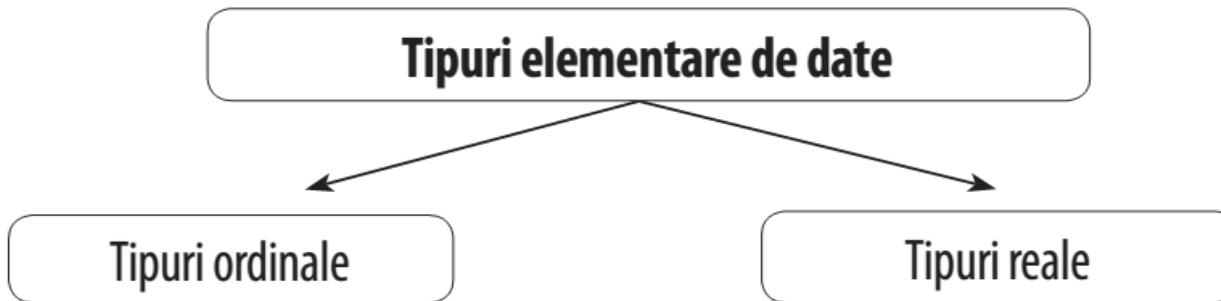


Tipurile elementare sunt tipuri atomare. Ele nu pot fi divizate în părți mai mici.

Structurile de date sunt combinații de alte tipuri, definite prin descrierea tipurilor componentelor și prin indicarea unor metode de structurare. Componentele tipurilor structurate pot fi elementare sau structurate.

Tipuri de date elementare

Clasificarea tipurilor de date elementare după continuitatea valorilor



Un **tip ordinal** definește o mulțime finită și ordonată de valori.

Caracteristicile tipului ordinal:

- definește o mulțime finită și ordonată de valori;
- are o valoare minimă;
- are o valoare maximă;
- fiecare valoare a tipului are asociat un număr ordinal;
- fiecare valoare, cu excepția valorii minime, are o valoare predecesor;
- fiecare valoare, cu excepția valorii maxime, are o valoare succesor.

Domeniul de definiție. Operații posibile

Fiecare valoare a unui tip ordinal are asociat un număr ordinal – poziția (0, 1, 2, ..., cu excepția tipurilor întregi).



Domeniul de definiție al tipului ordinal - valorile $V_0 \dots V_n$, unde V_0 - valoarea minimă, V_n - valoarea maximă.

Funcții aplicabile unei valori X de tip ordinal:

- **Succ(X)** – returnează succesorul lui X sau generează eroare;
- **Pred(X)** – returnează predecesorul lui X sau generează eroare;
- **Ord(X)** – returnează numărul de ordine asociat lui X în mulțimea valorilor corespunzătoare tipului ordinal al lui X.

Operatori relaționali

Forma de utilizare operatorilor relaționali:

<Operand_1> <Operator relational> <Operand_2>

- Operanzi trebuie să fie de același tip.
- Comparația se face între numerele lor de ordine.
- Rezultatul comparației este o valoare logică.

Operator	Exemplu	Comentariu
=	(A=B)	A este egal cu B
<>	(A<>B)	A nu este egal cu B
<	(A<B)	A este mai mic decât B
>	(A>B)	A este mai mare decât B
<=	(A<=B)	A este mai mic sau egal cu B
>=	(A>=B)	A este mai mare sau egal cu B

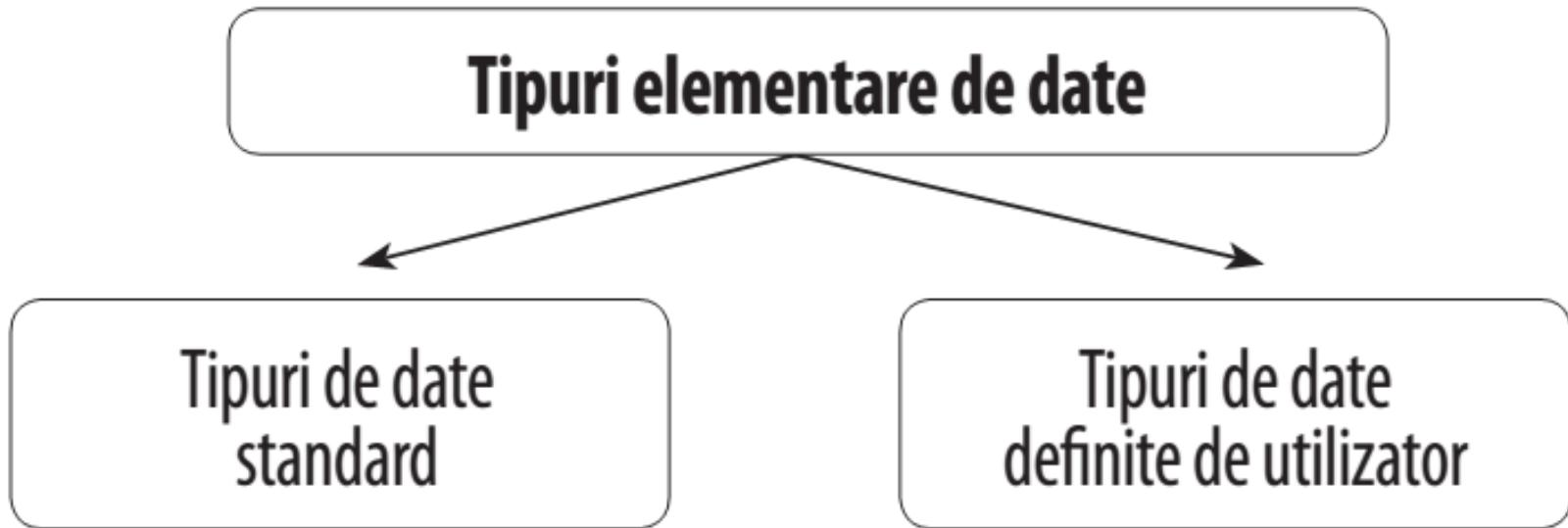
Tipul real de date

Un **tip real de date** definește o mulțime infinită de valori dintr-un interval dat.

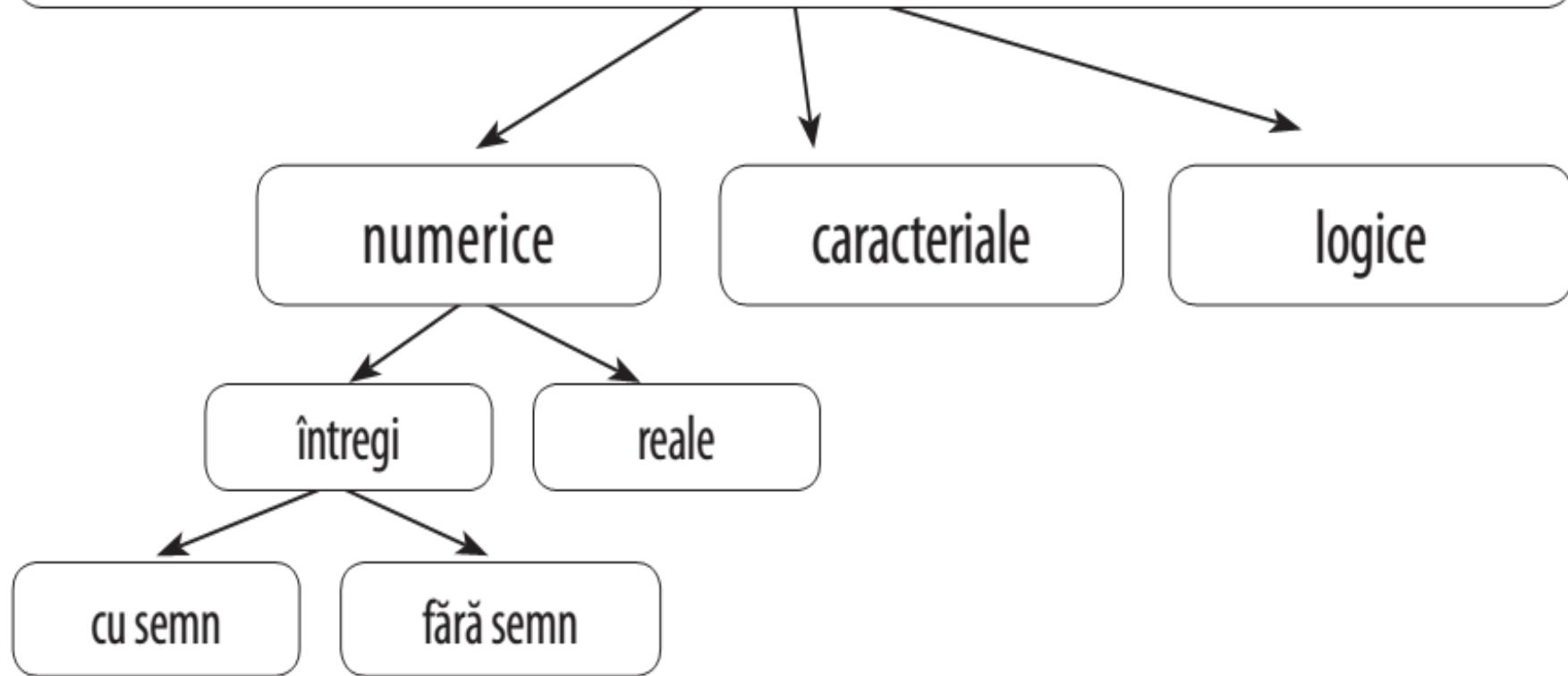
- Unei valori de tip real nu i se poate asocia valoarea-predecesor sau valoarea-succesor.
- Unei valori de tip real nu i se poate asocia un număr ordinal, care ar determina poziția valorii date în mulțimea infinită a valorilor din mulțimea tipului real.

Tipuri de date elementare

*Clasificarea tipurilor de date elementare după
necesitatea declarării*



Tipuri de date standard



Caracteristicele tipului de date

- **nume**, care reprezintă un identificator cu ajutorul căruia tipul se indică la declararea variabilelor;
- **domeniul valorilor** (domeniu de definiție) pe care le poate avea o variabilă de tipul respectiv;
- **operații posibile.**

Tipuri de date standard

Numele tipului de date	Descrierea tipului
<i>Integer</i>	tip de date numeric întreg cu semn
<i>Natural</i>	tip de date numeric întreg fără semn
<i>Real</i>	tip de date numere reale
<i>Char</i>	tip de date caracteriale
<i>Boolean</i>	tip de date logice

Tipul de date Integer

Datele de tip **Integer** reprezintă o submulțime a mulțimii numerelor întregi, care depinde de realizarea concretă a sistemului de calcul.

Caracteristicele tipului de date Integer:

- mulțime finită a numerelor întregi (de exemplu, datele de tip Integer se reprezintă pe 2 byți: interval -32768 ... +32767);
- valorile întregi se reprezintă ca numere întregi cu semn;
- valorile de tip Integer pot fi negative, nule, pozitive (de exemplu: -7, 4, 15, 10002).

Operatori aritmetici

Operatorii aritmetici	Comentariu
+	adunare
-	scădere
*	înmulțire
div	câtul întreg al împărțirii primului operand la al doilea
mod	restul împărțirii întregi a primului operand la al doilea

$$\text{Exemplu: } 7 + 2 = 9$$

$$7 - 2 = 5$$

$$7 \times 2 = 14$$

$$7 \text{ div } 2 = 3$$

$$7 \text{ mod } 2 = 1$$

$$\text{Ord}(4) = 4$$

$$\text{Ord}(-15) = -15$$

$$\text{Pred}(4) = 3$$

$$\text{Pred}(-4) = -5$$

$$\text{Succ}(4) = 5$$

$$\text{Succ}(-15) = -14$$

$$15 \text{ mod } 10 = 5$$

$$123 \text{ mod } 10 = 3$$

$$15 \text{ div } 10 = 1$$

$$123 \text{ div } 10 = 12$$

Exemple

$$1. \ 15 - (7 - 5)*3 =$$

$$10. \ 27 \bmod 2 =$$

$$2. \ \text{Pred}(-10) =$$

$$11. \ (2*3) \bmod (8 - 4) =$$

$$3. \ \text{Succ}(-5) =$$

$$12. \ 1245 \bmod 10 =$$

$$4. \ \text{Pred}(10) =$$

$$13. \ 1245 \bmod 10 =$$

$$5. \ \text{Succ}(-5) =$$

$$14. \ 1245 \bmod 100 =$$

$$6. \ \text{Ord}(10) =$$

$$15. \ 1245 \bmod 100 =$$

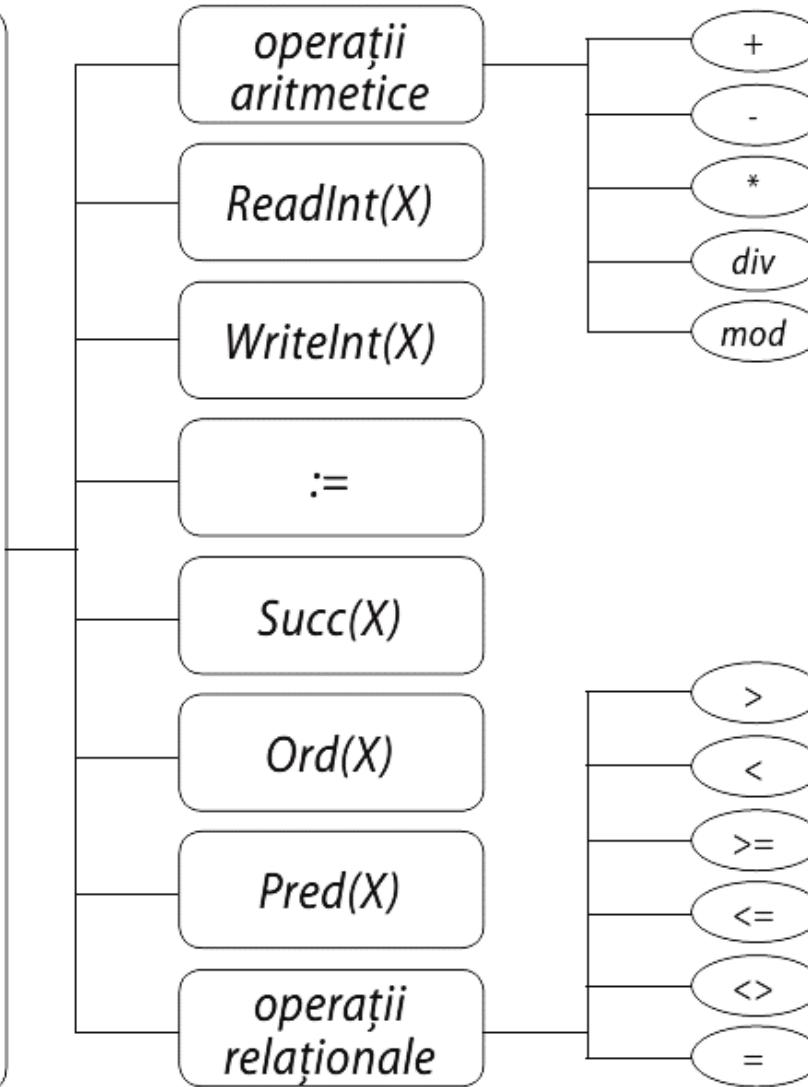
$$7. \ \text{Ord}(-5) =$$

$$8. \ 18 \bmod 5 =$$

$$9. \ 27 \bmod 3 =$$

Operații asupra datelor de tip Integer

Operații asupra datelor de tip Integer



Tipul de date Natural

Datele de tip **Natural** reprezintă o submulțime a mulțimii numerelor naturale.

Caracteristicele tipului de date Natural:

- mulțime finită a numerelor naturali (de exemplu, datele de tip Natural se reprezintă pe 2 byți: interval 0 ... 65535);
- valorile de tip Natural se reprezintă ca numere întregi fără semn;
- valorile de tip Natural pot fi numere pozitive sau nule.

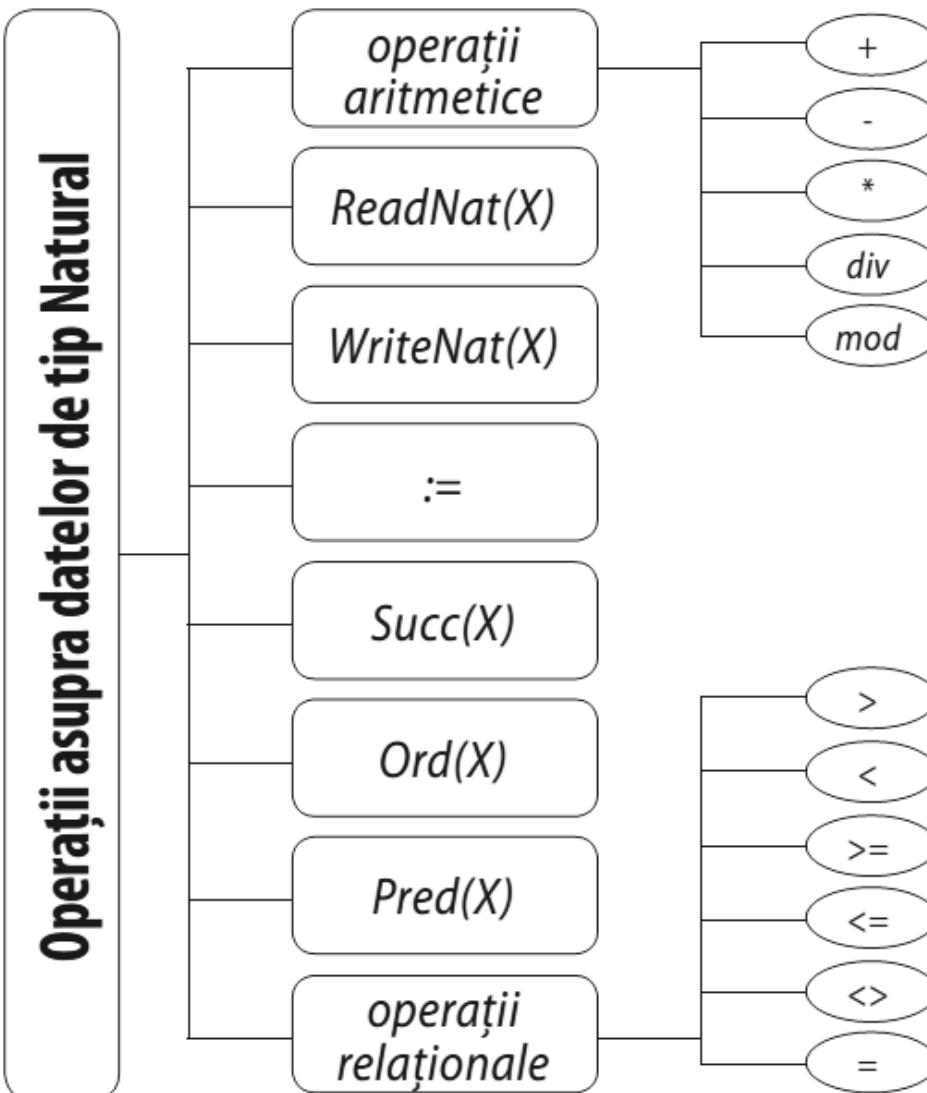
Operatori aritmetici

Operatorii aritmetici	Comentariu
+	adunare
-	scădere
*	înmulțire
div	câtul întreg al împărțirii primului operand la al doilea
mod	restul împărțirii întregi a primului operand la al doilea

Exemple:

$$5 + 3 = 8$$
$$5 - 3 = 2$$
$$5 * 3 = 15$$
$$5 \text{ div } 3 = 1$$
$$5 \text{ mod } 3 = 2$$
$$\text{Ord}(3) = 3$$
$$\text{Pred}(3) = 2$$
$$\text{Succ}(3) = 4$$

Operații asupra datelor de tip Natural



Tipul de date Real

Tipul de date **Real** se utilizează pentru memorarea valorilor reale și a valorilor întregi care nu pot fi reprezentate ca valori ale tipurilor Integer și Natural.

Caracteristicele tipului de date Real:

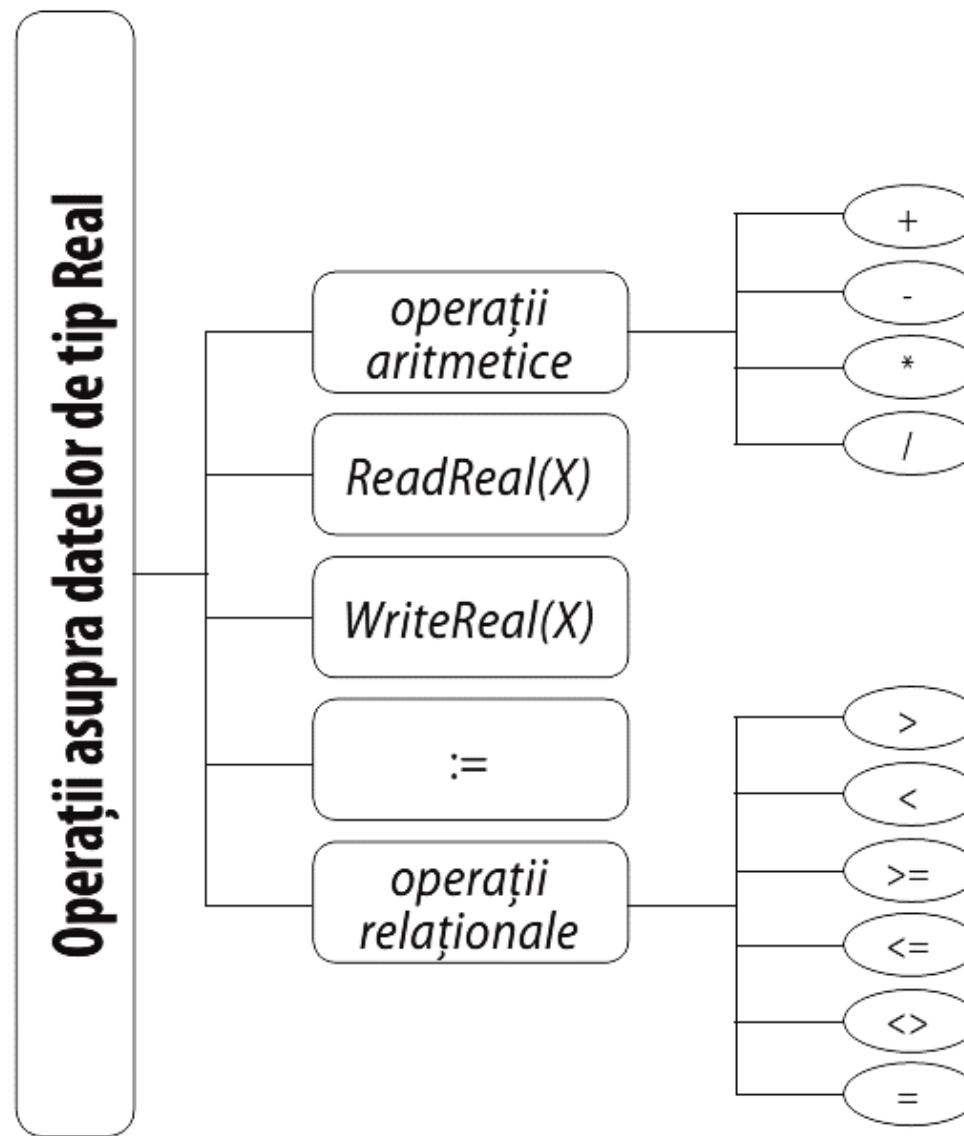
- datele de tip Real iau valori reale;
- operațiile cu date de tip Real produc rezultate aproximative.

Operatori aritmetici

Operatorii aritmetici	Comentariu
+	adunare
-	scădere
*	înmulțire
/	împărțire cu rezultat de tip real

Exemplu: $7 + 2 = 9$
 $7 - 2 = 5$
 $7 * 2 = 14$
 $7 / 2 = 3,5$

Operații asupra datelor de tip Real



Exemple. Tipuri elementare

1. Se consideră următorul fragment:

Var

a, b: Integer

Begin

a := 6

b := 7

a := (a+b+abs(a-b)) div 6

End

Indică valoarea variabilei a după executarea acestui fragment.

Exemple. Tipuri elementare

2. Se consideră următorul fragment:

Var

A, B, M, N, I: Integer

Begin

A := 2

B := 3+A

M := B div 2

I := 2

N := pred(M+I)

End

Care va fi valoarea variabilei N după executarea secvenței de mai sus?

Exemple. Tipuri elementare

3. Se consideră următorul fragment:

Var

N, S: Natural

Begin

N := 253

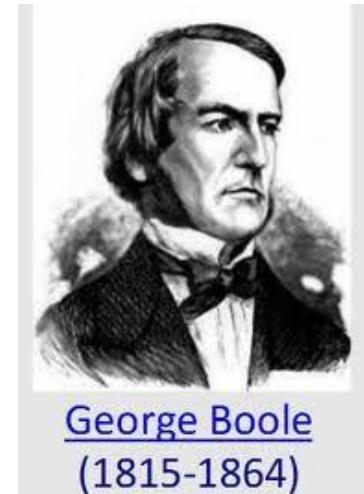
S := (N mod 10) + (N div 100) + ((N div 10) mod 10)

End

Indică valoarea variabilei S după executarea acestui fragment.

Tipul de date Boolean

Datele de tip **Boolean** pot avea valorile adevărat sau fals, desemnate prin constantele simbolice **True** și **False**.



George Boole
(1815-1864)

Caracteristicele tipului de date Boolean:

- datele de tip Boolean se reprezintă în memorie pe un byte (valoarea 0 pentru False și 1 pentru True);
- datele de tip Boolean nu se introduc de la tastatură și nu pot fi afisate;
- tipul de date Boolean este un tip ordinal.

Operatori logici

Operatorii aritmetici	Comentariu
Not	Negație logică
And	Și logic
Or	Sau logic
Xor	Sau exclusiv

Tabelul de adevăr al operației Not

X	Not X
<i>False</i>	<i>True</i>
<i>True</i>	<i>False</i>

Operator And

Tabelul de adevăr al operației And

X	Y	X and Y
False	False	False
False	True	False
True	False	False
True	True	True

Rezultatul operației And este True numai în cazul când ambii operanzi sunt egali cu True. Dacă măcar un operand este False, atunci rezultatul va fi False.

Operator Or

Tabelul de adevăr al operației Or

X	Y	X or Y
False	False	False
False	True	True
True	False	True
True	True	True

Rezultatul operației Or este True în cazul când măcar un operand este True. Rezultatul va fi False numai atunci când ambii operanzi au valoarea False.

Operator Xor

Tabelul de adevăr al operației Xor

X	Y	X xor Y
<i>False</i>	<i>False</i>	<i>False</i>
<i>False</i>	<i>True</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>True</i>
<i>True</i>	<i>True</i>	<i>False</i>

Rezultatul operației Xor este True atunci când operanții au valorile logice diferite și este False când ambii operanți au aceleași valori logice.

Operații asupra datelor de tip Boolean

E = X and (Y or X) xor (not Z)

Operații asupra datelor de tip Boolean

Valorilor de tip Boolean li se pot aplica **operatorii relaționali**.

Exemple:

1. $X=Y$ are valoarea True numai dacă X și Y au aceeași valoare de tip Boolean.
2. $X \leq Y$ are valoarea False numai dacă X este True și Y este False.

Pentru valori de tip Boolean pot fi aplicate funcțiile **Ord(X)**, **Pred(X)**, **Succ(X)**.

Exemple:

$$\text{Ord(False)}=0$$

$$\text{Ord(True)}=1$$

$$\text{Pred(True)}=\text{False}$$

$$\text{Succ(False)}=\text{True}$$

Expresii relaționale

Se folosesc următorii operatori relaționali:

$<$, \leq , $=$, $>$, \geq , \neq

Utilizarea acestor operatori se face sub forma:

$<\text{Operand}_1> <\text{Operator}> <\text{Operand}_2>$

Operanzi trebuie să fie de același tip.

Exemplu:

- $2 < 3$ returnează valoarea True;
- $0 > 7$ returnează valoarea False.

Expresiile relaționale pot să apară în instrucțiunile de atribuire.

Exemplu:

- $A := B > 10$
- Echilateral := $(A=B) \text{ And } (B=C)$

Var

Par: Boolean

n: natural

ReadNat (n)

Par:= n mod 2=0

Exemple. Expresii relaționale

1. Indică expresiile logice care exprimă următoarele afirmații.

1. Variabilele logice A și B au aceeași valoare. A. $\text{Ord}(A) + \text{Ord}(B) = 2$ B. $\text{Ord}(A) + \text{Ord}(B) = 0$ C. $\text{Ord}(A) + \text{Ord}(B) = 1$
2. Variabilele logice A și B au valori diferite D. $(\text{Ord}(A) + \text{Ord}(B) = 2) \text{ Or } (\text{Ord}(A) + \text{Ord}(B) = 0)$
3. Variabilele logice A și B conțin valoarea True.
4. Variabilele logice A și B conțin valoarea False.

2. Indică care va fi valoarea variabilei I după execuția fragmentului:

Var I: Integer

Begin

I := Ord(True Xor Succ(False)) + 7

End

Exemple. Expresii relaționale

3. Fie declarațiile:

Var

x, y : Natural

d : Boolean

Begin

$x := -1$

$y := 3$

$y := x + y$

$d := \text{Not}(x > y) \text{ Xor } (y > x)$

End

Determinați valoarea variabilei d.

Exemple. Expresii relaționale

4. Indică răspunsul care conține expresia logică care exprimă că x se află în interiorul segmentului $[-2,4]$.

- a) $(x > -2) \text{ Or } (x < 4)$
- b) $(x \geq -2) \text{ And } (x = < 4)$
- c) $(x > -2) \text{ Xor } (x < 4)$
- d) $(x \geq -2] \text{ And } (x = < 4]$
- e) Nici un răspuns nu este corect

Exemple. Expresii relaționale

5. Fie declarația:

Var

a, b: Integer

c: Boolean

Indică care dintre instrucțiuni este corectă.

- a) c := a + b
- b) a := c
- c) b := Ord(c) + a
- d) c := (a = b) and (a > 0)
- e) a := b + c
- f) c := a > b

Exemple. Expresii relaționale

6. Variabila întreagă n memorează un număr natural impar. Care dintre următoarele expresii are valoarea True?

- a) $\text{not}(n \bmod 2 <> 0)$
- b) $n \bmod 2 = 0$
- c) $n \bmod 2 <> 0$
- d) $\text{not}((n+1) \bmod 2 = 0)$

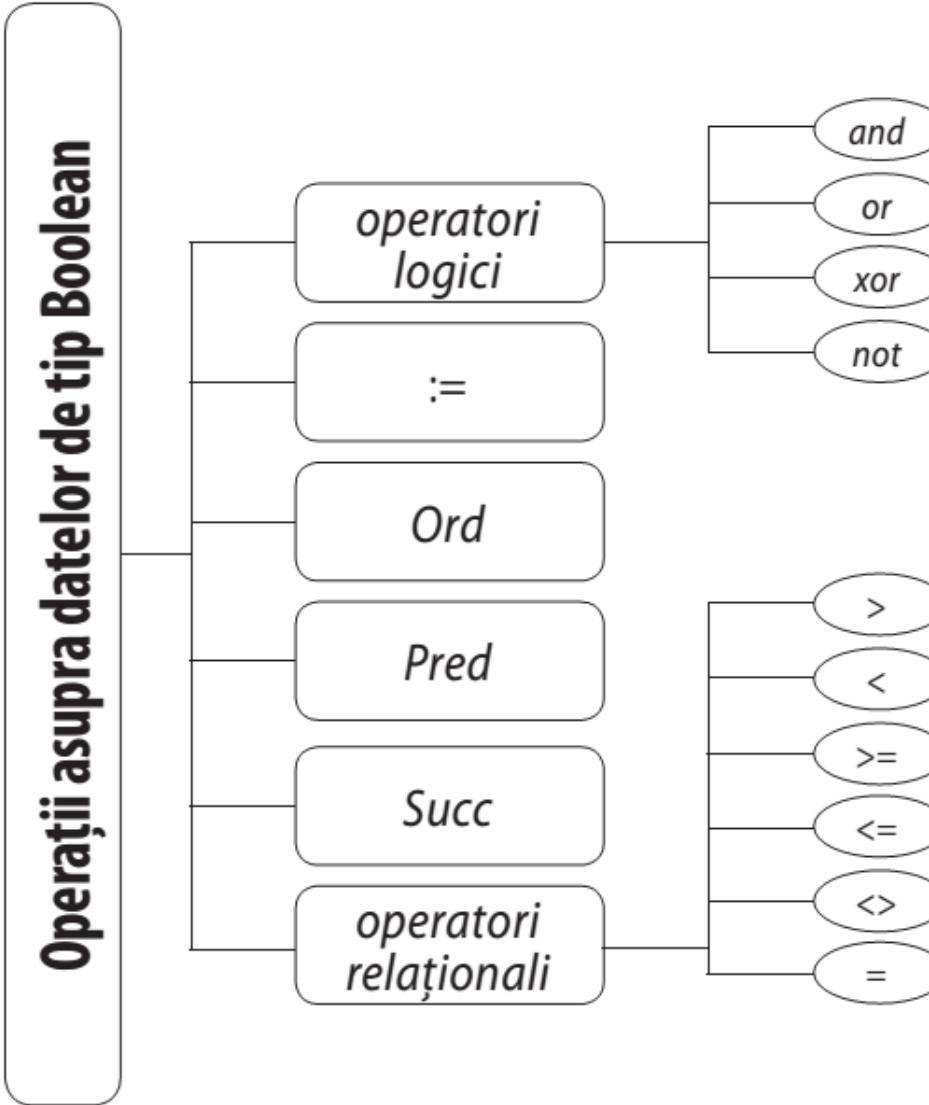
Exemple. Expresii relaționale

**7. Variabilele x, y, z și w sunt întregi,
x memorează valoarea 2,
y memorează valoarea 3,
z memorează valoarea 5,
iar w memorează valoarea 7.**

Care dintre următoarele expresii are valoarea True?

- a) $(y > z) \text{ or } (x > 3)$
- b) $(x = z) \text{ and } ((y = 3) \text{ or } (w = 7))$
- c) $(z \leq w) \text{ and } (x > 0) \text{ or } (y \geq x)$
- d) $(y \geq 3) \text{ and } (w < 7)$

Operații asupra datelor de tip Boolean



Tipul de date Char

O dată de tip **Char** poate lua ca valoare un singur caracter din setul extins de caractere ASCII (American Standard Code for Interchange Information).

ASCII conține codurile a 256 de caractere.

Caracteristicile tipului de date Char:

- datele de tip Char se reprezintă în memorie pe un byte, prin codul ASCII al caracterului corespunzător;
- tipul Char este un tip ordinal;
- pentru date de tip Char pot fi aplicate funcțiile ordinale Succ, Pred, Ord, Chr (inversă funcției Ord) și operațiile relaționale;
- valorile de tip Char pot fi citite de la tastatură și afișate pe ecran.

Alfabetul ASCII

sp	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
'	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
p	q	r	s	t	u	v	w	x	y	z	{	}	~		
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	

Tipul de date Char. Exemple

- Literele mari și mici ale alfabetului latin au coduri successive. Caracterul „A” are codul 65. Caracterul „a” are codul 97;
- Cifrele au coduri successive. Caracterul „0” are codul 48, „1” – 49;
- Succ('a') returnează caracterul 'b';
- Succ('4') returnează caracterul '5';
- Pred('C') returnează caracterul 'B';
- Ord('A') returnează 65, codul ASCII al caracterului 'A'
- Chr(65) returnează caracterul 'A';
- Chr(49) returnează caracterul '1';
- 'A'<'a' returnează valoarea True.

Tipul de date Char. Exemple

1. Se consideră următorul fragment:

Var

$A, B, C: Char$

$M: Natural$

Begin

$A := 'b'$

$B := succ(A)$

$C := succ('D')$

$M := ord(B) + ord(C)$

End

Care va fi valoarea variabilei M după executarea acestui fragment?

Tipul de date Char. Exemple

2. Se consideră următorul fragment:

Var

A, B, C: Char

M: Natural

Begin

A := 'B'

B := pred(A)

C := succ('c')

M := ord(B) - ord(C)

End

Care va fi valoarea variabilei M după executarea acestui fragment?

Citirea și afișarea valorilor de tip Char

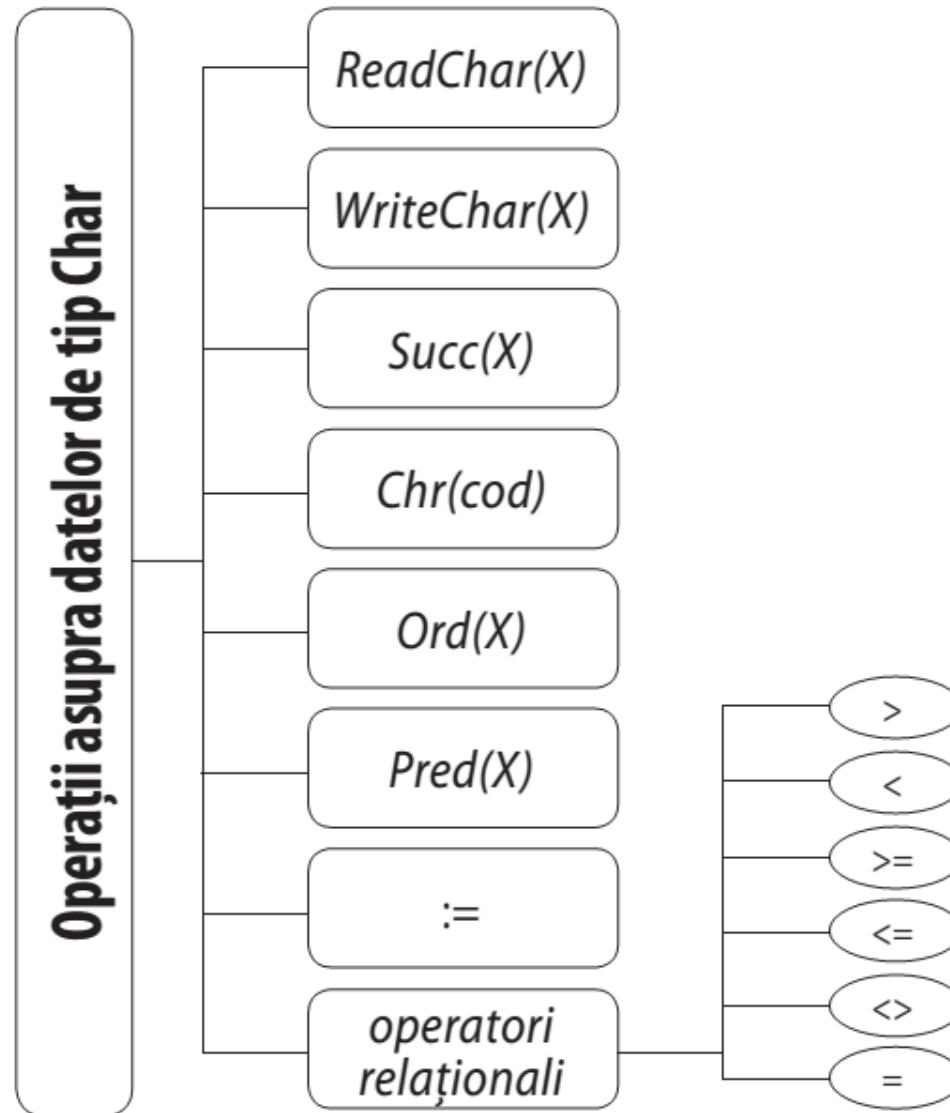
Citirea se face cu instrucțiunea **ReadChar(X)**, unde X este o variabilă de tip Char.

Afișarea valorilor de tip Char se face cu instrucțiunea **WriteChar(X)**, unde X este o expresie de tip Char sau cu instrucțiunea **WriteString(X)**, unde X este un sir de caractere.

Instrucțiunea **Writeln** realizează trecerea la începutul rândului următor.

Utilizarea instrucțiunii <i>WriteChar</i>	Utilizarea instrucțiunii <i>WriteString</i>
<i>WriteChar('P')</i> <i>WriteChar('a')</i> <i>WriteChar('c')</i> <i>WriteChar('e')</i>	<i>WriteString('Pace')</i>

Operații asupra datelor de tip Char



Transformarea tipurilor de date

Funcția	Destinația
$\text{IntToNat}(X)$	Transformă valoarea X de tip <i>Integer</i> în valoare de tip <i>Natural</i>
$\text{IntToReal}(X)$	Transformă valoarea X de tip <i>Integer</i> în valoare de tip <i>Real</i>
$\text{NatToInt}(X)$	Transformă valoarea X de tip <i>Natural</i> în valoare de tip <i>Integer</i>
$\text{NatToReal}(X)$	Transformă valoarea X de tip <i>Natural</i> în valoare de tip <i>Real</i>
$\text{RealToInt}(X)$	Transformă valoarea X de tip <i>Real</i> în valoare de tip <i>Integer</i>
$\text{RealToNat}(X)$	Transformă valoarea X de tip <i>Real</i> în valoare de tip <i>Natural</i>

Exemplu:

Var

- A: Natural
- B: Integer
- C: Real

B:=10

A:=IntToNat(B)

C:=IntToReal(B)

Sarcini pentru lucrul independent

1. Ce valoare va avea variabila a la sfârșitul următoarei secvențe de instrucțiuni?

Var a, b : Integer

$a := 3 \quad b := 7$

$b := a + b \text{ div } 2$

$a := a - b \text{ div } (2 * a)$

2. Fie a, b, c și d patru variabile reale. Care din următoarele instrucțiuni atribuie variabilei d media aritmetică a valorilor variabilelor a, b și c ?

a) $d := (a+b+c)/3$

b) $d := a/3+b/3+c/3$

c) $d := a+b+c/3$

d) $d := (a+b+c)/3-1$

Sarcini pentru lucrul independent

2. Ce va afișa următorul algoritm dacă se citește valoarea **2567**?

Var

a: Natural //Date de intrare
b: Natural // Date intermediare
c: Natural //Date de ieșire

Begin

Read (a)
b := a mod 100
a := a div 100
*c := b*100 + a*
Write (c)

End

Sarcini pentru lucrul independent

3. Ce va afișa următorul algoritm pentru valorile citite **7** și **23**?

Var

*a, b: Natural // Date de
intrare/ieșire*

Begin

Read(a, b)

a := a + b

b := a - b

a := a - b

Write(a, b)

End

Sarcini pentru lucrul independent

4. Se consideră următorul fragment?

Var

*x, y, z: Integer //Date de intrare
k: Boolean //Date de ieșire*

Begin

x := 1

y := 2

z := 3

k := Not ((X+Y) > 0) and Not (Z>0)

End

Sarcini pentru lucrul independent

1. Descrieți datele necesare pentru reprezentarea temperaturii zilei.
2. Descrieți datele necesare pentru reprezentarea unei date calendaristice. Scrieți fragmentul care reprezintă data de 15 martie 2012.
3. Descrieți datele necesare pentru reprezentarea mărimii salariului lunar al unui angajat.
4. Descrieți datele necesare pentru reprezentarea prețului a 3 produse și scrieți fragmentul care calculează suma necesară pentru procurarea acestor produse.

Informația pentru acces în sistemul Moodle

1. Adresa de acces:

<http://old.elearning.usarb.md/moodle>

2. Datele pentru intrare în sistemul Moodle:

The screenshot shows the Moodle login page with the following elements:

- Header:** Includes the university logo, navigation links for "Stiri", "Suport & asistență", and "Contacte".
- MENU PRINCIPAL (Left Sidebar):** Lists site navigation links: Pagina principală, Stiri site, Despre noi, Contacte, Ajutor, Răspunsuri la întrebări frecvente, Suport & asistență, and Cursuri.
- Content Area:** Features logos for TecTNet, Tempus, and the European Union, with the text "În parteneriat cu:". Below this is a sidebar titled "CATEGORII DE CURSURI" containing a list of course categories and their counts, many of which are collapsed (indicated by a triangle icon).
 - TecTNet (1)
 - Cursuri Teach Me (1)
 - TEREC (5)
 - CRUNT. Cursuri elaborate (6)
 - Cursuri elaborate în cadrul proiectului WETEN (2)
 - Cursuri elaborate în cadrul proiectul WETEN de către masteranzi (7)
 - Ciclul II (Masterat) (1)
 - Socioumane (4)
 - Stiinte Reale, Economice si ale Mediului (5)
 - Ciclul I (Licență) (14)
 - Socioumane (22)
 - Stiinte Reale (28)
- LOGARE (Right Sidebar):** Contains fields for "Utilizator" and "Parolă", a "Tine minte numele de utilizator" checkbox, and "Autentificare" and "Ați uitat parola?" buttons.
- Bottom Sidebar:** Displays the text "Platforma de învățare a Universității de Stat „Alecu Russo” din Bălți".
- MENU PRINCIPAL (Bottom Right Sidebar):** Lists site navigation links identical to the top sidebar.

Disciplina “Bazele programării”

Structura disciplinei "Bazele programării I":

Modulul 1.	Tipuri elementare de date și structuri de control	Unitatea didactică 1 - Unitatea didactică 8
Modulul 2.	Prelucrarea structurilor statice de date	Unitatea didactică 9 - Unitatea didactică 14

1 SEPTEMBRIE - 7 SEPTEMBRIE

Unitatea didactică 1 Etapele procesului de programare.

Finalități:

După studierea unității didactice 1 și realizarea sarcinilor de învățare propuse, veți deveni capabili:

- să explicați concepțele de bază ale procesului de programare (precizarea problemei, algoritm, proprietățile algoritmilor, formele de reprezentare a algoritmilor, etapele programării propriu-zise, tipurile de erori care pot apărea în dezvoltarea programelor);
- să realizați precizări ale problemelor reale și să evaluați calitatea acestora conform criteriilor predefinite.

Unitatea didactică 1. Etapele procesului de programare.

Etapele procesului de programare

Этапы процесса программирования

Resursa audio

Resursa video

Resursa audio-video

BOOK - Etapele procesului de programare

ПОСЛЕДНИЕ ДЕЙСТВИЯ

Действия с luni, 3 septembrie 2018, 21:03

Полный отчет о последних действиях

Со времени Вашего последнего входа ничего нового не произошло

НАСТРОЙКИ

Управление курсом

Оценки

▶ Банк вопросов

Переключиться к роли...

Вернуться к моей обычной роли

Настройки моего профиля

Администрирование