

1.4 UTILIZAREA FUNCTIILOR EXCEL

Procesorul de tabele Excel include un numar mare de functii predefinite (232), dar ofera si posibilitatea ca utilizatorul sa-si defineasca propriile functii, potrivit cerintelor de exploatare a aplicatiilor.

Funcțiile Excel permit efectuarea de calcule si prelucrari diverse, de la cele mai simple pâna la cele mai complexe.

1.4.1 FUNCTII PREDEFINITE

Funcțiile predefinite reprezinta formule speciale care respectând o anume sintaxa, executa operatii si prelucrari specifice, fiind destinate rezolvarii unor probleme si aplicatii ce contin elemente predefinite de calcul.

Unele functii predefinite sunt echivalente formulelor: de exemplu, formula de adunare a continutului celulelor A1, A2 si A4, adica =A1+A2+A4 este echivalenta cu functia =Sum(A1:A2;A4). Alte functii (majoritatea cazurilor) nu au echivalent in rândul formulelor, rezultatul scontat neputând fi obtinut decât prin aplicarea functiilor predefinite sau putând fi obtinut pe cale obisnuita, prin aplicarea succesiva a mai multor operatii si formule.

Folosirea functiilor predefinite este supusa unor reguli foarte stricte, a caror nerespectare poate conduce la un rezultat incorect sau generator de eroare.

Cea mai mare parte a functiilor predefinite au trei componente:

- semnul "egal"= (sau semnul "plus" +, pentru compatibilitate cu 1-2-3);
- numele functiei;
- unul sau mai multe argumente;

Nici un spatiu nu este admis ca separator intre cele trei componente ale functiilor predefinite. Argumentele se afla inchise intre paranteze rotunde si sunt separate printr-un separator zecimal. Acest separator poate fi virgula sau punct si virgula, dupa cum a fost configurat initial sistemul. In exemplele luate, se va lua in considerare ca separator zecimal caracterul "punct si virgula".

Exista si functii care nu au nevoie de precizarea argumentului, de exemplu:=NOW(), =TRUE(), =TODAY(), etc.

Exemplul urmator ilustreaza diferite argumente care se pot intâlni la o functie predefinita:

Funcție predefinită	Tip argument
=SUM(A2:A7)	plaja continua de celule
=SUM(A2:A7;A9;A11:A20)	plaja discontinua de celule
=MAX(59;36;84)	lista de valori
=DATE(62;10;18)	lista de valori data calendaristica
=IF(A1=A2;"Bun";Rau)	valoare logica
=INT(SUM(D1:D9))	funcție predefinită
=UPPER("Ionescu")	sir de caractere
=REPT("Ionescu",3)	sir si valoare numerica
=FACT(6)	valoare numerica

Excel accepta urmatoarele tipuri de argumente:

- **o conditie:** este o expresie logica care foloseste unul din operatorii logici =, <, >, <>, <=, >=, NOT(), AND(), OR() pentru o adresa de celula sau un nume de câmp. Conditia argumentului poate fi deci o formula, un numar, un nume de camp, un text. Functia evalueaza conditia si procedeaza la diferite operatii in functie de faptul daca conditia este adevarata sau falsa.

- **o locatie:** este o adresa, un nume de câmp, o formula sau functie care genereaza o adresa sau un nume de câmp.

- **un text:** orice secventa de caractere inclusa intre ghilimele, adresa sau un nume de câmp ce contine o eticheta tip sir de caractere sau o formula sau functie care returneaza o eticheta. Un sir de caractere folosit intr-o functie trebuie pus intre ghilimele pentru a nu fi confundat cu un nume de câmp.

- **o valoare:** un numar, adresa sau numele unei celule care contine un numar, o formula sau functie predefinita care returneaza un numar.

Toate tipurile de argumente pot fi folosite impreuna intr-o functie atunci când sintaxa este respectata.

O functie predefinita se poate introduce intr-o celula tastând-o ca atare (conform sintaxei) sau prin intermediul generatorului de functii.

Cea mai simpla metoda o reprezinta introducerea nemijlocita a functiilor predefinite, corespunzator sintaxei, in celula unde se va opera calculul respectiv (metoda recomandata).

In cel de-al doilea caz, se activeaza selectorul functiilor predefinite aflat pe bara de editare sau se activeaza comanda **Insert Function** (figura 1.52). Apoi, se alege functia respectiva, din caseta de dialog **Paste Function**, se valideaza si se completeaza sintaxa generata automat.

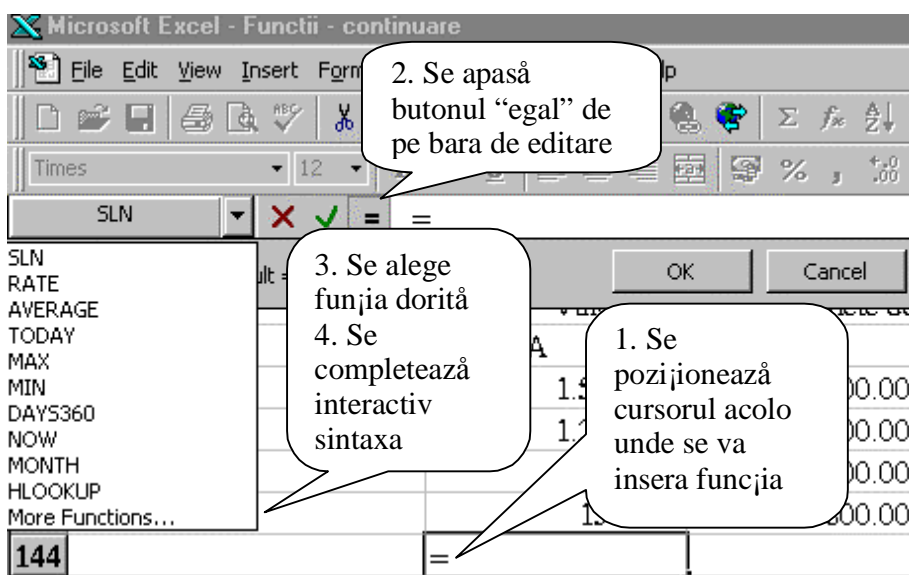


Fig 1.52 Etapele inserarii unei functii

Apelarea selectorului de functii se face prin apasarea butonului = (egal) aflat pe bara de editare, dupa care se deschide lista functiilor predefinite, se alege functia dorita, dupa care se completeaza interactiv argumentele.

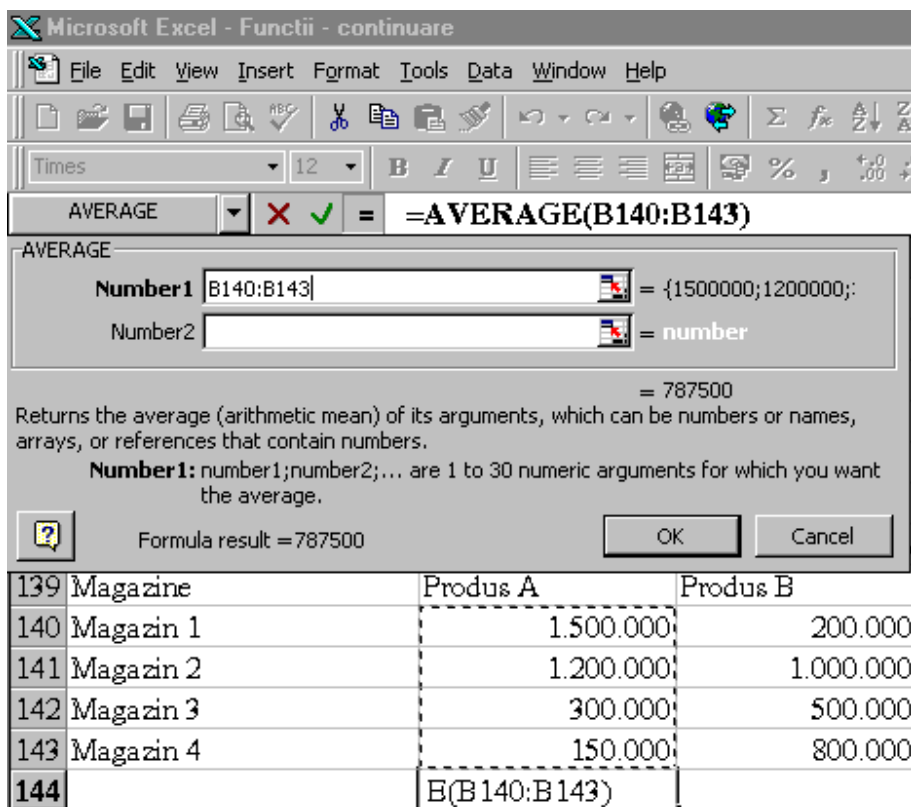


Fig. 1.53/1.54 Asistentul de functii/Exemple de date

Acest procedeu este prezentat in figura 1.53

Funcția poate fi aleasă din lista funcțiilor cele mai utilizate (**Most Recently Used**), din lista tuturor funcțiilor disponibile ordonate alfabetic (**All**), sau din categoriile de funcții specializate (**Financial, Date & Time, Math & Trig, Statistical ...**)

În celula din care s-a apelat funcția predefinită va apărea sintaxa funcției selectate și validându-se operația prin butonul **OK** se va genera rezultatul respectivei funcții.

Generatorul de funcții sau mai corect asistentul de funcții este prezentat în figura 1.53.

De regulă, utilizarea asistentului de funcții presupune parcurgerea a doi pași:

- *pasul 1* semnifică alegerea tipului de funcție;
- *pasul 2* presupune completarea interactivă a sintaxei funcției respective conform exemplului prezentat în figura 1.54.

Categorii de functii predefinite

Excel posedea un set impresionant de functii predefinite, in numar de 232, grupate pe tipuri potrivit utilitatii acestora la rezolvarea diferitelor probleme.

Astfel, consideram suficienta in rezolvarea aplicatiilor EXCEL, prezentarea a celor mai importante 99 de functii predefinite, grupate pe urmatoarele categorii (figura 1.55):

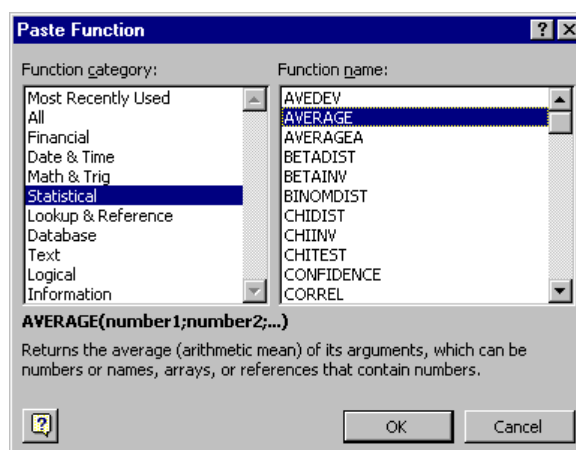


Fig. 1.55 Categorii de functii

1. **functii matematice si trigonometrice (Math & Trig)**: permit efectuarea de calcule matematice simple si complexe;
2. **functii statistice (Statistical)**: permit efectuarea unor calcule statistice utilizând serii de valori;
3. **functii de informare (Information)**: afiseaza informatii despre celule si câmpuri;
4. **functii logice (Logical)**: determina valoarea de adevar sau de fals - corespunzator unei conditii;
5. **functii baza de data (Database)**: efectueaza diferite calcule asupra unor rubrici, intr-o baza de date, corespunzator unor criterii definite;
6. **functii de cautare si consultare (Lookup & Reference)**: permit localizarea continutului unei celule;
7. **functii calendar sau data calendaristica (Date & Time)**: manipuleaza numere care reprezinta date calendaristice sau timp;
8. **functii text sau sir de caractere (Text)**: ofera informatii legate de textul existent in celule si permit operatii cu etichete;
9. **functii financiare (Financial)**: permit realizarea de calcule economico-financiare predefinite.

In continuare, prezentam cele mai importante functii predefinite, precizând ca cea mai mare parte a lor sunt perfect compatibile ca sintaxa si ca semnificatie cu functiile arond aferente procesorului de tabele LOTUS 1-2-3.

1.4.1.1 FUNCTIILE MATEMATICE SI TRIGONOMETRICE

Funcțiile matematice și trigonometrice (Math & Trig) permit efectuarea diferitelor calcule, de la cele mai simple la cele mai complexe, pentru rezolvarea de aplicații ce solicită instrumente matematice și trigonometrice de uz curent.

	A	B	C	D	E	F	G
60	Denumire produs	Data vânzării	Cantitate vândută	Preț unitar	Valoare	Comision 2%	
61	Produs A	01-Iul-98	125	56.000	7.000.000	140.000	
62	Produs B	01-Iul-98	250	31.000	7.750.000	155.000	
63	Produs C	01-Iul-98	300	25.000	7.500.000	150.000	
64	Produs A	02-Iul-98	200	56.000	11.200.000	224.000	
65	Produs B	03-Iul-98	180	31.000	5.580.000	111.600	
66	Produs C	03-Iul-98	156	25.000	3.900.000	78.000	
67	Sumă totală vânzări (câmp continuu)				=SUM(E61:E66)		
68	Sumă vânzări pe 01-Iul-98 și pe 03-Iul-98				=SUM(E61:E63;E65:E66)		
69	Sumă comision (câmp numit)				=SUM(Comision)		

Fig. 1.56 Funcția SUM

=SUM(**lista**) adună valorile dintr-o listă precizată ca argument.

Lista poate conține câmpuri continue sau discontinue referite prin adrese (coordonate) sau prin nume de câmp(uri).

Funcția de însumare este completată - spre ușurința utilizatorului - cu butonul **Auto Sum**. Funcția generată de butonul respectiv însumează pe linie sau pe coloană valori adiacente (valorile nu trebuie să fie întrerupte în succesiunea lor de celule vide sau de celule care să conțină texte). Auto-însumarea operează astfel pe linie sau pe coloană până acolo unde se întâlnește primul semn de discontinuitate (figura 1.56).

Pot exista mai multe cazuri (exemplificate în figura 1.56):

- se plasează cursorul acolo unde se dorește să se calculeze suma (eventual selectând o plajă de celule pe linie sau o coloană unde să se depună rezultatele
- calculelor) și se activează butonul **AutoSum** prin dublu-click;

	A	B	C	D	E
72	Denumire produs	Cantitate vândută	Preț unitar	Valoare	TOTAL
73	Produs A			7.000.000	1540000
74	Produs B			7.750.000	1705000
75	Produs C			7.500.000	1650000
76	Produs A			11.200.000	2464000
77	Produs B			5.580.000	1227600
78	Produs C	156	25.000	3.900.000	858000
79			TOTAL	=SUM(D73:D78)	

Fig. 1.57 Funcția AutoSum

- se selecteaza plaja de celule de insumat, inclusiv zona unde se vor plasa rezultatele insumarii (o linie mai jos si/sau o coloana mai la dreapta), dupa care se activeaza butonul **AutoSum** prin dublu-click.

=**PRODUCT (lista)** multiplica valorile continute intr-o lista. Un exemplu edificator este prezentat in figura 1.58.

=**SUBTOTAL(referinta-tip;câmp de regrupat)** calculeaza un rezultat ce provine dintr-o grupare a datelor operând diferite operatii specifice (conform referintelor-tip) asupra unui câmp de regrupat.

D	E	F
Valoare	TVA	Total
7.000.000	1540000	
7.750.000	1705000	
7.500.000	1650000	
11.200.000	2464000	
5.580.000	1227600	
3.900.000	858000	

D	E	F
Valoare	TVA	Total
7.000.000	1540000	1.540.000
7.750.000	1705000	1.705.000
7.500.000	1650000	1.650.000
11.200.000	2464000	2.464.000
5.580.000	1227600	1.227.600
3.900.000	858000	858.000
42.930.000	9.444.600	52.374.600

Fig. 1.58 Functia PRODUCT

Exemple de referinte-tip ar fi:

- | | | | | | |
|---|---------|---------|---|-------|--------|
| 1 | AVERAGE | Medie | 2 | COUNT | Numara |
| 4 | MAX | Maximum | | | |
| 5 | MIN | Minimum | | | |
| 6 | PRODUCT | Produs | | | |
| 9 | SUM | Suma | | | |

In exemplul prezentat in figura 1.59 se calculeaza suma (referinta-tip 9) valorilor produselor vândute pe 01-Iul-98 (câmpul de regrupat este E31:E33).

=**SUMPRODUCT(lista)** multiplica valorile situate in celulele corespondente, aferente unor serii de câmpuri, iar apoi aduna rezultatele obtinute. In exemplul prezentat in figura urmatoare se calculeaza prin functia SUMPRODUCT valoarea totala a vânzarilor, adica suma dintre produsele cantitatilor (C31:C36) si preturilor (D31:D36).

=**SUMIF(câmp de evaluat; criteriu; câmp de insumat)** aduna continutul celulelor potrivit unui criteriu dat.

In exemplul din figura 1.59 se calculeaza prin functia SUMIF, suma comisioanelor la vânzarile de produse (5% din valoare) pentru valorile vândute de peste 10.000.000 lei. In acest caz câmpul de evaluat reprezinta valoarea (E31:E36), criteriul este de tip text si anume ">10000000", iar câmpul de insumat este comisionul (F31:F36).

	A	B	C	D	E	F	G	H
30	Denumire produs	Data vânzării	Cantitate vândută	Pret unitar	Valoare	Comision 5%		
31	Produs A	01-Iul-98	150	50.000	PRODUCT(C31;D31)			
32	Produs B	01-Iul-98	350	31.000	7.750.000	132.125		
33	Produs C	01-Iul-98	400	35.000	7.500.000	150.000		
34	Produs A	02-Iul-98	200	56.000	11.200.000	224.000		
35	Produs B	02-Iul-98	180	31.000	5.580.000	111.900		
36	Produs C	03-Iul-98	150	25.000	3.900.000	78.000		
37					=SUMPRODUCT(C31:C36;D31:D36)			
38	Suma vânzări 01-Iul-98				SUBTOTAL(9;E31:E33)			
39	Suma comisioane pe vânzări >10.000.000 lei.					=SUMIF(E31:E36;">10000000";F31:F36)		

Fig. 1.59 Funcțiile SUMTOTAL, SUMPRODUCT, SUMIF

=ROMAN(numar;format) converteste numerele din format cifric arab in text ce semnifica numere cu format cifric roman. Formatul –cu valori de la 0 la 4- reprezinta gradul de concizie al numarului roman nou generat. Numarul arab de transformat trebuie sa fie intreg. Un exemplu de astfel de transformare este prezentat in figura 1.60.

=RAND() returneaza un numar aleator cuprins intre 0 si 1;

=ABS(numar) returneaza valoarea absoluta dintr-un numar;

=LN(numar) calculeaza logaritmul natural al unui numar specificat ca argument;

	A	B	C
12	Nr. în cifre arabe	Funcția	Rezultatul
13	1998	=ROMAN(A13;0)	MCMXCVIII
14	1998	=ROMAN(A14;1)	MLMVLIII
15	1998	=ROMAN(A15;2)	MXMVIII
16	1998	=ROMAN(A16;3)	MVMIII
17	1998	=ROMAN(A17;4)	MVMIII

Fig. 1.60 Funcția ROMAN

=LOG(numar;baza) returneaza logaritmul unui numar intr-o baza specificata;

	A	B
1	Funcția	Rezultatul
2	=SIN(20)	0,91294525
3	=COS(0)	1
4	=TAN(2)	-2,18503986
5	=ASIN(0,3)	0,30469265
6	=ACOS(0,3)	1,26610367
7	=ATAN(0,3)	0,29145679
8	=DEGREES(ATAN(0,3))	16,6992442
9	=RADIANS(180)	3,14159265

	A	B
1	Funcția	Rezultatul
2	=POWER(14;2)	196
3	=SQRT(196)	14

Fig. 1.61 Funcții trigonometrice și POWER și SQRT

=LOG10(numar) returneaza logaritmul in baza 10 dintr-un numar;

=EXP(X) calculeaza baza logaritmului natural ridicata la puterea X. Baza este o constanta si are valoarea 2,7182818....;

	A	B
20	Funcția	Rezultatul
21	=RAND()	0,870628572
22	=ABS(-453,12)	453,12
23	=LN(2,3)	0,832909123
24	=LOG(5)	0,698970004
25	=LOG10(120)	2,079181246
26	=EXP(5)	148,4131591
27	=MOD(3,4;2)	1,4
28	=FACT(4)	24

Fig. 1.62 Funcții matematice

=**MOD(X;Y)** calculeaza restul impartirii argumentului X la arg. Y;

=**FACT(numar)** calculeaza factorialul unui numar pozitiv;

=**POWER(numar;putere)** returneaza rezultatul unui numar ridicat la putere (figura 1.61);

=**SQRT(numar)** calculeaza radacina patrata a argumentului;

=**SIN(X)** returneaza valoarea argumentului X in radiani;

=**COS(X)** calculeaza cosinusul argumentului X in radiani ;

=**TAN(X)** calculeaza tangenta argumentului X in radiani;

=**ASIN(X)** calculeaza arc-sinusul argumentului X in radiani (similar =**ACOS(X)** si =**ATAN(X)**);

=**DEGREES(unghi)** converteste radianii in grade;

=**RADIANS(unghi)** converteste grade in radiani;

=**ROUND(X,numar de zecimale)** rotunjeste argumentul numeric X la un numar specificat de zecimale;

	A	B
1	Funcția	Rezultatul
2	=ROUND(196,278;2)	196,28
3	=ROUND(PRODUCT(125,3698;36,7892);1)	4612,3
4	=PI()	3,1415927
5	=ROUND(PI();3)	3,142
6	=INT(PI())	3

Fig. 1.63 Funcția de rotunjire

=**PI()** returneaza valoarea numarului PI;

=**INT(numar)** afiseaza partea intrega a argumentului (a numarului real), fara a-l rotunji.

1.4.1.2 FUNCTIILE STATISTICE

Funcțiile statistice (Statistical) permit efectuarea de calcule statistice utilizând serii de valori:

=MAX(lista) returnează cea mai mare valoare din lista. Lista poate fi compusă din: numere, formule numerice, adrese sau nume de câmpuri;

=MIN(lista) returnează cea mai mică valoare din lista;

=AVERAGE(lista) calculează media valorilor din lista;

=GEOMEAN(lista) calculează media geometrică a valorilor dintr-o listă

=HARMEAN(lista) calculează media armonică a valorilor dintr-o listă;

=MEDIAN(lista) calculează valoarea mediană dintr-o listă;

=COUNT(lista) numără celulele ocupate dintr-o listă de câmpuri;

Exemplul din figura 1.64 ilustrează utilizarea funcțiilor statistice prezentate:

	A	B	C	D
138		Valoare vânzări pe puncte de desfacere		
139	Magazine	Produs A	Produs B	Produs C
140	Magazin 1	1.500.000	200.000	600.000
141	Magazin 2	1.200.000	1.000.000	850.000
142	Magazin 3	300.000	500.000	900.000
143	Magazin 4	150.000	800.000	450.000
144				
145	Funcția		Rezultatul	
146	=MAX(B140:B143)		1.500.000	
147	=MIN(B140:B143;D140:D143)		150.000	
148	=AVERAGE(C140:C143)		625.000	
149	=GEOMEAN(B140:B143)		533483,823	
150	=HARMEAN(ProdusB)		432432,4324	
151	=MEDIAN(B140:D143)		700.000	
152	=COUNT(B140:B143)		4	

Fig. 1.64 Funcții statistice

1.4.1.3 FUNCTIILE DE INFORMARE

Funcțiile de informare (Information) afișează informații referitoare la celule și câmpuri:

=ISBLANK(X) determină dacă X sau amplasamentul definit de argumentul X este sau nu o celulă vidă. Funcția returnează TRUE -valoarea logică de adevăr- dacă amplasamentul este o celulă vidă și FALSE -valoarea logică de fals- în caz contrar;

=ISNUMBER(X) verifică dacă X conține o valoare numerică. Funcția returnează TRUE -adevărat- dacă X conține un număr, altfel returnează FALSE sau fals. Argumentul X poate fi o valoare, o adresă, text sau o condiție);

=ISTEXT(X) verifică dacă X conține un șir de caractere, returnând după caz TRUE sau FALSE.

=ISNONTEXT(X) verifică dacă X nu conține un șir de caractere, returnând după caz TRUE sau FALSE.

=ISLOGICAL(X) verifică dacă argumentul X conține o valoare de tip logic returnând după caz TRUE sau FALSE.

=ISERROR(X) verifică dacă argumentul X conține o valoare de tip eroare, returnând după caz TRUE sau FALSE.

O parte din funcțiile de informare sunt exemplificate împreună cu funcțiile logice.

1.4.1.4 FUNCTIILE LOGICE

Funcțiile logice (Logical) determină evaluarea unor expresii și în funcție de acestea furnizează acțiuni sau rezultate complexe, generând valori de adevăr sau de fals - corespunzător unor condiții (acestea pot fi evaluate și în lanțuite cu ajutorul operatorilor logici AND, OR, NOT).

=IF(condiție;X;Y) testează argumentul condiție și în funcție de rezultatul evaluării logice, generează argumentul X dacă condiția este adevărată sau argumentul Y dacă aceasta este falsă.

Argumentele X sau Y pot fi valori, șiruri de caractere (plasate între ghilimele), nume de câmpuri sau adrese de celule sau câmpuri care conțin aceste valori. În locul argumentelor X sau Y se pot imbrica alte structuri condiționale IF, generându-se potrivit condițiilor ulterioare, X1,Y1 sau X2,Y2 și așa mai departe.

=AND(evaluare logica1,evaluare logica2,...) returnează valoarea logică TRUE dacă toate argumentele sunt adevărate și valoarea logică FALSE dacă unul sau mai multe argumente sunt false;

=OR(evaluare logica1,evaluarea logica2,...) returnează valoarea logică TRUE dacă orice argument este adevărat și valoarea logică FALSE dacă toate argumentele sunt false;

=NOT(evaluarea logica) inversează valoarea argumentului, returnând după caz TRUE sau FALSE;

=TRUE() returnează valoarea logică TRUE;

=FALSE() returnează valoarea logică TRUE;

Pentru exemplificarea functiei logice IF, furnizam urmatoarea aplicatie pentru calculul impozitului pe salariile colaboratorilor angajati cu Conventie Civila de Prestari Servicii:

Astfel, daca salariul brut este sub 500.000 lei, impozitul este de 10% din brut, altfel, daca salariul brut este cuprins intre 500.000 lei si 1.500.000 lei, impozitul este de 50.000 lei + 20% din ceea ce depaseste 500.000 lei salariu brut, daca salariul brut este cuprins intre 1.500.000 lei si 2.500.000 lei, impozitul este de 250.000 lei + 25% din ceea ce depaseste 1.500.000 lei salariu brut, daca salariul brut este cuprins intre 2.500.000 lei si 3.500.000 lei, impozitul este de 500.000 lei + 30% din ceea ce depaseste 2.500.000 lei salariu brut, daca salariul brut depaseste 3.500.000 lei, impozitul este de 800.000 lei + 40% din ceea ce depaseste 3.500.000 lei salariu brut.

Aplicatia este astfel construita incat sa exemplifice (didactic) toate functiile logice (figura 1.65, 1.66).

Intr-un prim pas s-a construit o coloana de "Evaluare logica" care returneaza in functie de un test facut asupra salariului brut, valoarea logica de fals (FALSE) daca salariul brut este text, blank sau este mai mic ca zero si returneaza valoarea logica de adevar (TRUE) in caz contrar.

c2		=IF(OR(ISTEXT(B2);ISBLANK(B2);B2<=0);FALSE();TRUE())						
	A	B	C	D	E	F	G	H
	Nume	Salariul brut	Evaluare logica	Impozit				
1								
2	Popescu Dan	-3.000	FALSE	0				
3	Popa Florin	aaa	FALSE	0				
4	Nedelcu Maria		FALSE	0				
5	Albu Constanta	0	FALSE	0				

Fig. 1.65 Functii logice (I)

Intr-un al doilea pas se calculeaza impozitul pe salarii daca coloana "Evaluare logica" (s-a utilizat functia NOT()) nu contine valoarea logica FALSE.

In aplicatia de mai sus s-a construit o structura conditionala imbricata, unde s-a exemplificat intr-o ramura IF si functia logica AND.

B2		=IF(AND(C2=TRUE(),(B2<=500000),(B2>500000),(B2<=1500000),(B2>1500000),(B2<=2500000),(B2>2500000),(B2<=3500000),(B2>3500000)),(B2<=500000),(10%*B2),(50000+(B2-500000)*20%),(B2<=1500000),(250000+(B2-1500000)*25%),(B2<=2500000),(500000+(B2-2500000)*30%),(B2>3500000),(800000+(B2-3500000)*40%))						
	Salariul brut	Evaluare logica	Impozit					
2	500000	TRUE	50000					
3	1700000	TRUE	300000					
4	2800000	TRUE	650000					
5	4500000	TRUE	1340000					

Fig. 1.66 Functii logice(II)

1.4.1.5 FUNCTIILE BAZA DE DATE

Funcțiile baza de date (Database) returnează acțiuni - valori sau etichete (suma; medie; maxim; minim; caută; număr) dintr-un câmp de date - corespunzător unei baze de date, după o anumită rubrică, conform unui criteriu de selecție

Funcțiile tip baza de date au în mod invariabil aceeași listă de argumente:

- - *baza de date*: reprezintă tabelul Excel sub forma unui câmp de date, de unde informația va fi consultată sau extrasă;

- - *rubrică*: semnifică atributul sau proprietatea asupra căreia operează calculul făcut de funcția tip baza de date. Rubrică poate fi identificată prin numele sau prin numărul de ordine al acesteia în cadrul bazei de date;

- - *câmp de criterii*: reprezintă unul sau mai multe câmpuri continue în care se pot preciza restricțiile, care se regroupează în criterii de selecție la care trebuie să răspundă interogarea respectivă.

Principalele funcții tip baza de date sunt:

=DSUM(baza de date;rubrică/nr.rubrică;câmp de criterii) returnează suma valorilor unei rubrici aferente unei baze de date, care răspunde unui criteriu de selecție;

=DMAX(baza de date;rubrică/nr.rubrică;câmp de criterii) returnează cea mai mare valoare dintr-o rubrică aferentă unei baze de date, corespunzător unui criteriu de selecție;

	A	B	C	D	E	F	G	H
4	Centralizatorul facturilor emise							
5								
6	Cod Client	Nume Client	Adresa	Localitate	Nr factura	Data facturii	Valoare	Platit
7	100	Client1	Str. Florilor	Pitești	456	09.28.97	1.700.000	NU
8	101	Client2	Str. Sincal	București	888	01.05.96	850.000	DA
9	102	Client7	Str. Averescu	Brasov	343	01.14.98	2.000.000	NU
10	101	Client1	Str. Florilor	Pitești	87	02.24.98	1.100.000	DA
11	104	Client4	Str. Politehnica	Ploiești	566	03.12.98	900.000	NU
12	105	Client3	Str. 9 Mai	București	70	04.15.98	340.000	NU
13	104	Client4	Str. Politehnica	Ploiești	244	02.15.98	3.600.000	NU
14	107	Client10	Str. Plopilor	Iasi	670	11.17.97	2.500.000	DA
15	101	Client2	Str. Sincal	București	122	04.12.98	760.000	DA
16	100	Client1	Str. Florilor	Pitești	406	03.20.98	1.000.000	NU
17	102	Client7	Str. Averescu	Brasov	111	05.03.97	2.300.000	DA

Fig. 1.67 Aplicație pentru funcțiile baza de date

=DMIN(baza de date;rubrică/nr.rubrică;câmp de criterii) returnează cea mai mică valoare dintr-o rubrică aferentă unei baze de date, corespunzător unui criteriu de selecție;

=DAVERAGE(baza de date;rubrică/nr.rubrică;câmp de criterii) calculează media valorilor unei rubrici aferente unei tabelă - pentru o baza de date, potrivit criteriului de selecție specificat

=DCOUNT(baza de date;rubrică/nr.rubrică;câmp de criterii) numără celulele ocupate într-o tabelă baza de date, conform unor criterii specificate;

=DGET(baza de date;rubrica/nr.rubrica;câmp de criterii) returneaza continutul unei rubrici pentru o baza de date, corespunzator unui criteriu specificat. Functia este utila pentru a regasi o informatie unica; Exemple edificatoare de utilizare a functiilor tip baza de date sunt ilustrate in figura urmatoare, urmând ca alte aplicatii mai complexe sa fie prezentate in detaliu in capitolul ce trateaza bazele de date create si exploatate sub Excel.

Pornind de la un tabel definit pe coordonatele A6:H17, considerat a fi o baza de date care repertoriaza facturile emise de o firma catre clientii sai, se pot pune in evidenta cu ajutorul functiilor tip baza de date, informatii calculate potrivit unor interogari specifice.

Prezentam in figurile 1.67- 1.68 câteva exemple de utilizare a functiilor tip baza de date, utilizând câmpuri de criterii definite de utilizator potrivit unor cerinte de interogare.

1.4.1.6 1.4.1.6 FUNCTIILE DE CAUTARE SI CONSULTARE

Funcțiile de cautare si consultare (Lookup & Reference) permit cautarea, identificarea si referirea continutului unor celule:

	A	B	C	D	E	F	G
18	FORMULA				REZULTATUL	CRITERII	
19	Suma valorilor facturilor plătite din Bucuresti					Localitate	Platit
20	=DSUM(A6:H17;7;F19:G20)				1.610.000	Bucuresti	DA
21	Maximul valoric al facturilor neîncasate din Ploiesti					Localitate	Platit
22	=DMAX(A6:H17;"Valoare";F21:G22)				3.600.000	Ploiesti	NU
23	Minimul valoric al facturilor emise						
24	=DMIN(A6:H17;7;F23:F24)				340.000		
25	Media facturilor emise și neîncasate					Platit	
26	=DAVERAGE(A6:H17;"Valoare";F25:F26)				1.590.000	NU	
27	Numărul facturilor emise în București și Ploiești					Localitate	
28	=DCOUNT(A6:H17;7;F27:F29)				5	Bucuresti	
29						Ploiesti	
30	Care este valoarea facturii numărul 670					Nr factura	
31	=DGET(A6:H17;"Valoare";F30:F31)				2500000	670	

Fig. 1.68 Modul de utilizare a funcțiilor baza de date

=CHOOSE(index-numeric;lista de valori) returneaza in urma unei alegeri dintr-o lista de valori, o actiune sau o valoare, ce urmeaza a fi activata sau executata, corespunzator unui index numeric. Indexul numeric determina care valoare (de tip text, numerica sau referinta celulara) din lista de argumente va fi selectata. Indexul este un numar cuprins intre 0 si 29.

=COLUMN(referinta celulara sau câmp) returneaza numarul colanei corespunzatoare referintei celulare sau numarul primei coloane pentru câmpul specificat;

=COLUMNS(câmp) returneaza numarul de coloane aferente câmpului specificat ca argument;

=ROW(referinta celulara sau câmp) returneaza numarul liniei corespunzatoare referintei celulare sau numarul primei linii a câmpului specificat ca argument;

=ROWS(câmp) returneaza numarul de linii pe care il ocupa câmpul specificat ca argument;

=AREAS(referinta celulara) indica numarul de zone contigue dintr-un câmp. Daca respectivul câmp contine mai multe zone contigue, atunci argumentul se mai inchide intr-o paranteza suplimentara;

In figura 1.69 sunt prezentate mai multe exemple de utilizare a functiilor enumerate mai sus.

	A	B	C	D
120	Magazine	Produs A	Produs B	Produs C
121	Magazin 1	1.500.000	200.000	600.000
122	Magazin 2	1.200.000	1.000.000	850.000
123	Magazin 3	300.000	500.000	900.000
124	Magazin 4	150.000	800.000	450.000
125				
126	Functia			Rezultatul
127	=CHOOSE(2;"Produs A";"Produs B";"Produs C")			Produs B
128	=AVERAGE(CHOOSE(2;B121:B124;C121:C124;D121:D124))			625.000
129	=AVERAGE(B121:CHOOSE(2;B124;C124;D124))			706.250
130	=ROW(B121:D124)			121
131	=ROWS(B121:D124)			4
132	=COLUMN(B121:D124)			2
133	=COLUMNS(B121:D124)			3
134	=AREAS(B121:D124)			1
135	=AREAS((B120:D120;A121:A124;B121:D124))			3

Fig. 1.69 Aplicatie pentru functiile de cautare si consultare

=VLOOKUP(cheie;câmp de consultare;coloana de recuperat) returneaza continutul unei celule ce figureaza intr-o coloana dintr-un tablou de consultare verticala.

Sintaxa functiei de consultare verticala admite trei argumente si anume:

-- *cheie*: reprezinta valoarea dupa care are loc cautarea sau consultarea, (adresa absoluta/relativa sau nume de câmp);

-- *câmp (sau tabel) de consultare*: este câmpul asupra caruia opereaza consultarea prin cautarea valorii cheii precizate anterior;

-- *coloana de recuperat*: este numarul coloanei (numerotarea incepe cu 1) de unde va fi recuperata informatia gasita in tabelul de consultare, corespunzator valorii cheii de cautare.

In mod obligatoriu tabelul de consultare va fi sortat crescator dupa coloana care contine valorile cheii de consultare (comanda **Data Sort**, iar in rubrica **Sort by** se va preciza numarul sau numele coloanei dupa care se va face sortarea)

=HLOOKUP(cheie;câmp de consultare;linie de recuperat) returneaza continutul unei celule ce figureaza intr-o anumita linie a unui tablou de consultare orizontala.

Argumentul cheie (sub forma unei referinte celulare sau nume de câmp) va fi cautat in prima linie a câmpului de consultare, iar daca valoarea va fi gasita pe un numarul de linie precizat de ultimul argument, valoarea respectiva va fi returnata de functia HLOOKUP.

In mod obligatoriu tabelul de consultare orizontala trebuie sortat dupa valorile crescatoare ale cheii de consultare aflate in prima linie (sortare de la stânga la dreapta). Daca valorile cheii nu sunt sortate, se va selecta tabelul de consultare si se va activa comanda de sortare (de la stânga spre dreapta): **Data Sort**, butonul **Option** si din rubrica **Orientation** se alege optiunea **Sort left to right**.

Daca informatia cautata in tabelul de consultare verticala sau orizontala nu va fi gasita, se va returna cea mai apropiata valoare (pe verticala sau pe orizontala) de cheia de consultare.

Pentru exemplificarea celor doua functii de consultare propunem urmatoarea **aplicatie**:

O societate comerciala de distributie intocmeste, cu ajutorul procesorului de tabele EXCEL, *facturi pentru livrarile efectuate*.

Optional, respectiva societate efectueaza si transportul marfii comandate la domiciliul clientului, firma practicând tarife diferite in functie de cantitatea transportata (in tone) si de orasul de destinatie.

Tarifele de transport sunt grupate intr-un tablou in functie de destinatie (prima linie) si de cantitatea transportata (prima coloana). Tabloul care urmeaza a fi considerat tabel de consultare orizontala a fost definit pe coordonatele F20:J28 (figura 1.70) si a fost in prealabil sortat de la stânga la dreapta dupa prima linie, adica dupa destinatie.

	E	F	G	H	I	J
20	Destinația/ Cantitate transport	Brasov	Bucuresti	Ploiesti	Predeal	Sinaia
21	1	75.000	15.000	30.000	60.000	50.000
22	2	125.000	30.000	60.000	120.000	100.000
23	3	175.000	45.000	90.000	155.000	150.000
24	4	225.000	60.000	120.000	210.000	200.000
25	5	275.000	75.000	150.000	260.000	250.000
26	6	325.000	90.000	180.000	310.000	300.000
27	7	375.000	105.000	210.000	360.000	350.000
28	8	425.000	120.000	240.000	410.000	400.000

Fig. 1.70 Date pentru aplicatia de cautare.

Firma isi are inregistrati clientii intr-o baza de date (definita pe coordonatele E1:I7) (figura 1.71) care regroupeaza elementele de identificare ale acestora (“Client”, “Adresa”, “Localitate”, “Cod fiscal”, “Cont bancar”).

In egala masura exista si o alta baza de date –definita pe coordonatele A20:C28 (figura 1.72) sub forma unui nomenclator de preturi pentru fiecare produs in parte. Cele doua baze de date sunt sortate dupa valorile crescatoare ale primei coloane si contin informatii pertinente ce concura la realizarea automata a facturii.

	E	F	G	H	I
1	Client	Adresa	Localitate	Cod fiscal	Cont bancar
2	Euro97	Pacuri nr 66	Campina	123000111	123.456
3	Sageata	19 Mai nr 5, sector 6	Bucuresti	124000222	456.789
4	Star	Ceasornicului nr 5, secto	Bucuresti	126000444	789.456
5	Terom	Parcul cu castani nr 8	Ploiesti	150000555	147.258
6	Teski	Mihail Moxa nr. 3-5	Bucuresti	234000666	258.369
7	Zodiac	Aleea Dumbravita nr 1	Bucuresti	345000777	369.147

Fig. 1.71 Date pentru aplicatia de cautare

	A	B	C
20	Cod produs	Denumire produs	Pret
21	99	Produs1	10500
22	101	Produs2	50000
23	102	Produs3	120000
24	103	Produs4	16000
25	104	Produs5	11700
26	105	Produs6	48000
27	106	Produs7	10000
28	107	Produs8	15500

Fig. 1.72 Nomenclatorul de preturi

Factura procesata cu Excel are urmatoarea forma (figura 1.73):

	A	B	C	D	E	F	G	H	I
1	Factura nr:	378378	din	04.Mar.98					
2									
3	Client	Euro97							
4	Adresa	Facut nr 36							
5	Localitate	Campina							
6	Cod fiscal	123.000.111							
7	Cont bancar	123.456							
8	Transportul este facut de furnizor?		da						
9									
10	Adaos comercial		25%		Destinatia	Brasov			
11									
12	Cod produs	Denumire produs	Cant. livrata	Cheltuieli transport	Prez	Valoare	Majorari	TVA	Valoare factura
13	99	Facut nr 1	8	42000	10000	520000	130000	80000	912.960 L
14	100	Facut nr 2	3	10000	10000	200000	50000	40000	455.000 L
15	105	Facut nr 3	2	12000	48000	240000	60000	66000	478.240 L
16	120	ERR	3	10000	15000	200000	50000	80000	455.000 L
17	Total factura	378378		90000	22375	1241250	744750	240000	2.306.320 L
18	Nr. produse	4							

Fig. 1.73 Factura obtinuta

Utilizatorul va introduce prin tastare, pentru completarea facturii doar *denumirea clientului*, *codul produsului facturat*, *cota de adaos comercial*, *cantitatea livrata*, iar optional daca se doreste sau nu transport, precum si *destinatia transportului*. In rest toate operatiile sunt facute automat cu ajutorul formulelor si a functiilor Excel.

Factura se proceseaza in mod obisnuit, incepând a se calcula intr-un prim timp "Valoarea", "Majorarile", "TVA-ul" si "Valoarea facturata". La calculul "Valorii" se va lua in calcul si o cota variabila de adaos comercial (celula D10 a fost fixata cu adresa absoluta -\$D\$10- pentru a nu se decala la copierea formulei ce calculeaza valoarea), precum si cheltuielile de transport. "Majorarile" de intârziere se pot calcula pe transe, prin structuri conditionale imbricate. "TVA"-ul reprezinta 22% din "Valoare" + "Majorari", iar "Valoarea facturii" reprezinta suma dintre "Valoare", "Majorari" si "TVA".

Intr-un al doilea timp se pot calcula totalurile pe rubricile procesate anterior utilizând clasica functie SUM.

Interesante de prezentat sunt *facilitatile de consultare verticala si orizontala*.

La tastarea numelui de client in celula C3, se vor recupera automat dintr-un tabel de consultare verticala (definit anterior pe coordonatele E1:I7), informatiile legate de acest identificator si anume: "Adresa", "Localitatea", "Codul fiscal" si "Contul bancar".

Astfel in celula C4 s-a scris formula de consultare verticala (VLOOKUP) (figura 1.74) pentru recuperarea adresei clientului, anume: "se cauta cheia de consultare (celula \$C\$3-Client) in tabelul de consultare definit pe coordonatele \$E\$1:\$I\$7 si in caz ca valoarea este gasita, se va recupera informatia din coloana 2, corespunzatoare cheii de consultare".

Coordonatele cheii si tabelului de consultare au fost blocate prin utilizarea de adrese absolute pentru ca formula ce contine consultarea verticala sa poata fi copiată fara ca respectivele coordonate sa se decaleze.

B	C	D	E
8888	din	04.Mar.98	Client
			Euro97
Client	Euro97		Sageata
Adresa	=VLOOKUP(\$C\$3,\$E\$1:\$I\$7,2)		
Localitate	Campina		Terom
Cod fiscal	123.000.111		Teski
Cont bancar	123.456		Zodiac

Fig. 1.74 Functia de consultare verticala

Cheia de consultare fiind in acest caz de tip text nu trebuie sa aiba valori vide si nici numerice. Pentru aceasta, procedura de consultarea verticala a fost completata cu teste facute asupra celulei care contine cheia de consultare (C3). Daca cheia are valoarea vida "ISBLANK(C3) sau (OR()) daca contine o valoare alta decât text "ISNONTEXT(C3), atunci se va afisa un spatiu (" "), altfel se va face consultarea verticala.

In aceste conditii, consultarea verticala va avea urmatoarea forma:

=IF(OR(ISBLANK(C3);ISNONTEXT(C3));" ";VLOOKUP(C3;\$E\$1:\$H\$7;3)), fapt
 ilustrat si in figura 1.75.

	B	C	D	E	F	G	H	
1	8888	din	04.Mar.98	Client	Adresa	Localitate	Cod fiscal	C
2				Euro97	Pacuri nr 66	Campina	123000111	
3	Client	Euro97		Sageata	19 Mai nr 5	Bucuresti	124000222	
4	Adresa	Pacuri nr 66		Star	Ceasornic	Bucuresti	126000444	
5	Localitate	=IF(OR(ISBLANK(C3);ISNONTEXT(C3));" ";VLOOKUP(C3;\$E\$1:\$H\$7;3))						
6	Cod fiscal	123.000.111		Teski	Mihail Mox	Bucuresti	234000666	

Fig. 1.75 Functia de consultare verticala

Daca se tasteaza un client care nu exista in nomenclatorul de clienti (in tabelul de consultare verticala), functia VLOOKUP nu va semnala lipsa informatiei din tabel ci va returna informatia legata de cea mai apropiata valoare a cheii de consultare. De exemplu, daca s-ar introduce clientul cu numele "Sarmis", se vor recupera prin VLOOKUP informatiile aditionale corespunzatoare celei mai apropiate valori ale cheii, adica informatiile legate de clientul "Star". Functia VLOOKUP nu va semnala inexistenta cheii de consultare "Sarmis". Pentru inlaturarea acestui neajuns, procedura de consultare verticala a fost completata cu un test de existenta a cheii ce consultare in tabelul de consultare".

Acest test de existenta verifica daca valoarea cheii de consultare este gasita in prima coloana a tabelului de consultare. Daca valoarea respectiva exista in tabel inseamna ca s-a gasit cheia de consultare si in consecinta consultarea verticala se va efectua returnând un rezultat corect, altfel se va afisa spatiu sau zero (ultimul caz folosindu-se daca celula respectiva participa ulterior la calcule) sau un mesaj de genul "cheie inexistentă".

Formula de testare a existentei cheii de consultare in tabel este urmatoarea:

IF(VLOOKUP(C3;\$E\$1:\$I\$7;1)<>C3;" ";VLOOKUP(C3;\$E\$1:\$I\$7;4)).

Rubricile: "Adresa" –C4-, "Localitatea" –C5-, "Cod fiscal" –C6-, "Cont bancar" –C7- se vor recupera prin acelasi procedeu de consultare verticala, recuperându-se dupa caz, prin functia VLOOKUP continutul coloanelor 2, 3, 4, si 5, corespunzator valorilor cheii de consultare declarate la adresa C3 (figura 1.76).

	B	C	D	E	F	G	H	I
1	8888	din	04.Mar.98	Client	Adresa	Localitate	Cod Fiscal	Cont bancar
2				Euro97	Pacuri nr 66	Campina	123000111	
3	Client	Euro97		Sageata	19 Mai nr 5	Bucuresti	124000222	
4	Adresa	Pacuri nr 66		Star	Ceasornic	Bucuresti	126000444	
5	Localitate	Campina		Teski	Mihail Mox	Bucuresti	234000666	
6	Cod Fiscal	=IF(OR(ISBLANK(C3);ISNONTEXT(C3));" ";IF(VLOOKUP(C3;\$E\$1:\$I\$7;1)<>C3;" ";VLOOKUP(C3;\$E\$1:\$I\$7;4)))						
7	Cont bancar	VLOOKUP(C3;\$E\$1:\$I\$7;4))						

Fig. 1.76 Consultare verticala.

In mod asemanator se procedeaza si cu a doua consultare verticala, anume: in momentul tastarii "Codului de produs" este consultat vertical tabelul "PRETURI" declarat la adresa A20:C28, si daca in tabelul respectiv este gasita cheia de consultare "Cod produs" - se vor recupera automat: continutul coloanei 2 si 3 din tablou, adica "Denumire produs" si "Pret".

	A	B					
	Cod produs	Denumire produs	Cant. livrata	Cheltuieli transport	Pret	Valoare	Majorari TVA
13	99						
14	107						

Fig. 1.77 Aplicatie de consultare verticala

In figura 1.77 este prezentata procedura completa (cu teste facute asupra celulei ce contine cheia de consultare si cu test de existenta a valorii cheii in tabelul de consultare) de extragere a denumirii produsului, prin consultare verticala. Similar se procedeaza pentru extragerea pretului din tablou, corespunzator valorilor luate de codul produsului.

Consultarea orizontala a tabelului declarat pe coordonatele \$F\$20:\$J\$28 are loc dupa valorile luate de cheia de consultare - \$G\$10 "Destinatia". Daca cheia este gasita in tablou, se va recupera numarul de linie care va contine valoarea cheltuielilor de transport corespunzatoare destinatiei specificate.

In exemplul prezentat in figura 1.78, s-a operat un mic artificiu, anume "Cantitatea livrata" coincide logic cu numarul de linie de recuperat orizontal din tablou (astfel, nu s-a precizat numarul liniei recuperate, ci celula care contine livrata, aflata la adresa C13). Datorita faptului ca procedura de consultare orizontala este operationala incepând cu linia 1 (care contine invariabil titlurile rubricilor aferente destinatiei), celula "Cantitatea livrata" va indica numarul liniei de recuperat si va avea valoarea incrementata cu o unitate pentru a exista o concordanta intre valorile luate de aceasta si numarul liniei de recuperat. Daca nu s-ar fi operat acest artificiu, numarul liniei de recuperat ar fi decalat cu o unitate (adica, daca celula C13 - "Cantitatea livrata" - ar fi avut valoarea 3, s-ar fi recuperat linia numarul 3 din tabel -prima linie contine titlul rubricilor-, adica valoarea cheltuielilor de transport aferente pentru 2 tone transportate)

Prin functia HLOOKUP s-a consultat deci respectivul tablou, cautându-se valoarea luata de "Destinatia" in celula \$G\$10, recuperându-se numarul de linie ce corespunde logic cu "Cantitatea livrata".

Tabloul de consultare trebuie in mod obligatoriu sortat alfabetic dupa prima linie a sa.

Procedura de consultare orizontala poate fi completata si astfel imbunatatita (celula D14) prin urmatoarele teste:

- un test facut asupra celulelor ce contin: "Destinatia" (sa nu fie valoare vida sau numerica) si "Cantitatea livrata" (sa nu fie valoare de tip text, vida sau zero):

=IF(OR(ISNONTEXT(\$G\$10),ISBLANK(\$G\$10),ISTEXT(C14),ISBLANK(C14),C14=0;0;IF(.....) . Daca cel puțin unul din argumente este adevarat, functia va returna valoarea zero, altfel se vor testa si alte conditii de indeplinit;

5.9									
B	C								
8	Adresa furnizor de furnizori	d							
9									
10	Adresa comercial	25%	Destinatia	Brasov					
11									
12	Denumire produs	Cant. livrata	Cheltuieli transport	Pret	Valoare	Majorari	TVA	Valoare factura	
13	99 Produs1	8	425000	10500	530000	318000	69960	917.960 LEI	
14	107 Produs8	3	175000	15500	233125	139875	82060	455.060 LEI	
15	105 Produs6	2	125000	48000	245000	147000	86240	478.240 LEI	
16	125 ERR	3	175000	0	175000	105000	61600	341.600 LEI	
17	Total factura	8888	900000	18500	1183125	709875	299860	2.192.860 LEI	
18	nr. produs	4							
19									
20	Cod produs	Denumire produs	Pret	Destinatia/Cantitate	Brasov	Bucuresti	Ploiesti	Predeal	Sinaia
21	99	Produs1	10500	1	75000	15000	30000	60000	50000
22	101	Produs2	50000	2	125000	30000	60000	120000	100000
23	102	Produs3	120000	3	175000	45000	90000	150000	150000
24	103	Produs4	16000	4	225000	60000	120000	210000	200000
25	104	Produs5	11700	5	275000	75000	150000	260000	250000
26	105	Produs6	48000	6	325000	90000	180000	310000	300000
27	106	Produs7	10000	7	375000	105000	210000	360000	350000

Fig. 1.78 Aplicatie de consultare orizontala

- un test de existenta a "Destinatiei" (celula \$G\$10) in prima linie a tabloului de consultare orizontala: =IF(OR(...;\$G\$10<>HLOOKUP(\$G\$10;\$F\$20:\$J\$28;1));0;.....). Daca "Destinatia" este inexistentă in tablou, functia va returneaza valoarea zero, altfel se procedeaza la consultarea propriu-zisa;

- un test de existenta a "Cantitatii livrate" in prima coloana a unui tablou de consultare verticala: =IF(OR(C14<>VLOOKUP(C14;\$E\$20:\$E\$28;1);.....). Daca nu exista valoarea unei cantitati livrate in tabloul de consultare verticala definit pe coordonatele \$E\$20:\$E\$28, functia returneaza zero, altfel se procedeaza la consultarea propriu-zisa: (.....HLOOKUP(\$G\$10;\$F\$20:\$J\$28;C14+1));

- procedura ar putea fi completata si cu un test de efectuare a transportului: astfel, daca transportul este facut de furnizor (celula \$D\$8 are valoarea "da"), atunci se procedeaza la testele de mai sus si se executa in final consultarea orizontala, altfel cheltuielile de transport vor fi zero.

Figura 1.79 indica corespondentele create intre diferitele câmpuri, in procesul de consultare verticala si orizontala.

A	B	C	D	E	F	G	H	I	J
1	Factura nr.	8888	din	04.Mar.98	Client	Adresa	Localitate	Cod fiscal	Cont bancar
2					Euro97	Pacuri nr 66	Campina	123000111	123.456
3					Sageata	9 Main 5, sector 6	Bucuresti	124000232	456.789
4	Client	Euro97			Star	Ceasornicului nr 5, sed	Bucuresti	126000444	789.456
5	Adresa	Pacuri nr 66			Terom	Parcul cu castani nr 8	Ploiesti	150000555	147.258
6	Localitate	Campina			Teski	Mihail Moza nr. 3-5	Bucuresti	234000666	258.369
7	Cod fiscal	123.000.111			Zodiac	Aleea Dumbravita nr 1	Bucuresti	345000777	369.147
8	Cont bancar	123.456							
9	Transportul este facut de furnizor?	da							
10	Adresa comercial	25%			Destinatia	Brasov			
11									
12	Cod produs	Denumire produs	Cant. livrata	Cheltuieli transport	Pret	Valoare	Majorari	TVA	Valoare factura
13	99	Produs1	8	425000	10500	530000	318000	69960	917.960 LEI
14	107	Produs8	3	175000	15500	233125	139875	82060	455.060 LEI
15	105	Produs6	2	125000	48000	245000	147000	86240	478.240 LEI
16	125	ERR	3	175000	0	175000	105000	61600	341.600 LEI
17	Total factura	8888	900000	18500	1183125	709875	299860	2.192.860 LEI	
18	nr. produs	4							
19									
20	Cod produs	Denumire produs	Pret	Destinatia/Cantitate	Brasov	Bucuresti	Ploiesti	Predeal	Sinaia
21	99	Produs1	10500	1	75000	15000	30000	60000	50000
22	101	Produs2	50000	2	125000	30000	60000	120000	100000
23	102	Produs3	120000	3	175000	45000	90000	150000	150000
24	103	Produs4	16000	4	225000	60000	120000	210000	200000
25	104	Produs5	11700	5	275000	75000	150000	260000	250000
26	105	Produs6	48000	6	325000	90000	180000	310000	300000
27	106	Produs7	10000	7	375000	105000	210000	360000	350000

Fig. 1.79 Consultarea verticala si orizontala. Corespondente

1.4.1.7 1.4.1.7 FUNCTIILE TIP DATA CALENDARISTICA SI ORA

Funcțiile tip data calendaristică și ora (Date & Time) manipulează și operează calcule cu valori numerice ce reprezintă date calendaristice sau timp:

=NOW() returnează un număr corespunzător datei curente - cu zecimale ce reprezintă ora;

=TODAY() returnează un număr-data corespunzător datei curente;

=DATEVALUE("sir de caractere") calculează numărul-data corespunzător sirului de caractere în format data calendaristică (sirul trebuie plasat între ghilimele);

=DATE(an;luna;zi) calculează numărul-data pentru data calendaristică specificată ca argument;

=YEAR(numar-data) returnează corespunzător anului, un număr cuprins între 0 (1900) și 199 (2099) - extragând rezultatul dintr-un număr-data;

=MONTH(numar-data) extrage luna dintr-un număr-data, sub forma de valori cuprinse între 1 și 12;

=DAY(numar-data) generează un număr corespunzător zilei cu valori între 1 și 31;

=WEEKDAY(X) returnează numărul zilei din săptămâna corespunzător argumentului X care poate fi de tip număr data calendaristică sau text în format data calendaristică;

=DAYS360(data debut;data sfârșit) calculează numărul de zile între două date calendaristice considerând anul ca având 360 de zile;

=TIME(ora;minut;secunda) calculează un număr-timp corespunzător orei, minutului și secunde;

=TIMEVALUE("sir de caractere") returnează numărul-timp corespunzător sirului de caractere specificat în format data/ora (între ghilimele);

=HOUR(numar-timp) extrage ora dintr-un număr-timp (0,000000 pentru ora 24:00:00 și 9,999988426 pentru ora 23:59:59), sub forma unui număr cuprins între 0 și 23;

=MINUTE(numar-timp) extrage minutul dintr-un număr-timp, sub forma unui număr întreg cuprins între 0 și 59;

=SECOND(numar-timp) extrage secunda dintr-un număr-timp sub forma unui număr întreg cuprins între 0 și 59;

Un exemplu edificator de utilizare a functiilor de tip data calendaristica si ora este prezentat in figura 1.80.

	A	B	C
42	Funcția	Rezultatul	
43	=NOW()	26.07.1998 15:18	
44	=TODAY()	26.07.1998	
45	=DATEVALUE("27-Iun-98")	35973	27-Iun-98
46	=DATE(98;6;27)	35973	27-Iun-98
47	=YEAR(TODAY())	1998	
48	=MONTH(DATEVALUE("27-Iun-98"))	6	
49	=DAY(DATE(98;6;27))	27	
50	=WEEKDAY(DATE(98;6;27))	7	
51	=DAYS360(B45;B44)	29	
52	=TIME(14;35;0)	2:35 PM	
53	=TIMEVALUE("2:35 PM")	0,607638889	2:35:00 PM
54	=HOUR(TIME(14;35;0))	14	
55	=MINUTE(NOW())	18	
56	=SECOND(TIMEVALUE("23:26:04"))	4	

Fig. 1.80 Funcțiile tip data si ora

1.4.1.8 FUNCTIILE TEXT SAU SIR DE CARACTERE

Funcțiile text (Text): permit diferite operatii cu siruri de caractere si furnizeaza in egala masura informatii legate de textul existent in celule:

=**CHAR(cod numeric ASCII)** returneaza caracterul corespunzator codului numeric ASCII specificat ca argument;

=**TRIM(text)** afiseaza sirul de caractere specificat ca argument in care toate spatiile inutile sunt anulate (cu exceptia spatiilor care separa cuvintele textului);

=**CODE(text)** returneaza codul numeric pentru primul caracter din textul specificat ca argument;

=**CONCATENATE(text1;text2;...)** concateneaza mai multe siruri de caractere specificate ca argumente, intr-unul singur;

=**EXACT(text1;text2)** verifica daca doua siruri de caractere sunt identice. Comparând cele doua siruri, functia returneaza valoarea logica TRUE daca acestea sunt identice sau valoarea logica FALSE in caz contrar;

=**UPPER(text)** afiseaza cu majuscule textul specificat ca argument;

=**LOWER(text)** afiseaza cu minuscule textul specificat ca argument;

=**MID(text;N;X)** afiseaza X caractere ale textului specificat ca argument, incepând cu pozitia "N";

=**LEN(text)** returneaza numarul caracterelor ce formeaza textul specificat ca argument;

=**SUBSTITUTE(text-sursa;N;X;text-nou)** returneaza un nou sir de caractere (text-nou) la a "N"-a pozitie a textului-sursa, dupa ce au fost anulate X caractere;

=**REPT(text;numar de ori)** repeta afisarea textului de un numar specificat de ori;

=**PROPER(text)** determina scrierea cu majuscula a fiecărei prime litere din textul specificat ca argument;

	B	C
8	Funcția	Rezultatul
9	=CHAR(70)	F
10	=CHAR(70)&CHAR(73)	FI
11	=TRIM("Informatică de Gestiune")	Informatică de Gestiune
12	=CODE("Informatică de Gestiune")	73
13	=CONCATENATE("Informatică"," de ","Gestiune")	Informatică de Gestiune
14	=EXACT("Informatică","informatică")	FALSE
15	=UPPER("excel '97")	EXCEL '97
16	=LOWER("excel '97")	excel '97
17	=MID("Excel",3,2)	ce
18	=LEN("Microinformatică")	16
19	=REPT("la ",4)	la la la la
20	=PROPER("informatică generală")	Informatică Generală
21	=VALUE("25-Feb-98")	35851
22	=DOLLAR(1450,25)	1.450,25 LEI
23	=FIND("N","IONESCU",3)	3

Fig. 1.81 Exemple de funcții text

=**VALUE(text)** converteste un text ce reprezinta un numar intr-o valoare numerica (numarul ce figureaza in textul respectiv, trebuie sa corespunda unuia din formate numerice consacrate);

=**DOLLAR(numar;zecimale)** converteste un numar in text, folosind un format monetar;

=**FIND(text1;text2;N)** localizeaza pozitia la care incepe textul1 in textul2 incepând cautarea cu pozitia N;

Exemplificarile functiilor de tip text sau sir de caractere se gasesc prezentate in figura 1.81.

1.4.1.9 FUNCTIILE FINANCIARE

Funcțiile financiare (Financial) efectuează o serie de calcule economico-financiare furnizând prin valorile returnate informații utile referitoare la amortismente, la rentabilitatea investițiilor, plasamentelor, împrumuturilor etc.

=PV(rata dobânzii;numar de perioade;marimea platii;[valoare viitoare;tipul]) returnează valoarea actuală (present value) aferentă unei sume investite sau depozitate la banca, prin plăți periodice, în condițiile unei rate constante a dobânzii.

Funcția financiară PV calculează deci valoarea prezentă a unei sume investite, adică valoarea curentă a unei serii de plăți viitoare. Funcția se utilizează pentru a se determina dacă valoarea de revenire a unei anumite investiții este favorabilă sau nu, ținând cont de costul inițial al investiției.

Funcția PV (ca și alte funcții financiare PMT, FV) este considerată a fi o funcție-anuitate, adică operează cu o investiție sau un depozit la care toate plățile sunt egale și sunt efectuate la intervale regulate.

Argumentele funcției PV au următoarea semnificație:

- - *rata dobânzii* reprezintă procentul de dobândă perceput pentru o anumită perioadă;
- - *numar de perioade* reprezintă numărul total de plăți periodice;
- - *marimea platii* semnifică valoarea plății făcute în fiecare perioadă;
- - *valoare viitoare* reprezintă suma totală care se dorește a fi realizată după ultima plată;
- - *tipul* este un parametru care semnifică faptul că plata se face la începutul perioadei (valoarea 1) sau la sfârșitul perioadei (valoarea 0 – implicită).

Dacă sunt omise ultimele două argumente, acestea vor fi considerate ca având valori nule.

Argumentele rata dobânzii și numărul de perioade trebuie exprimate în aceeași unitate de timp – luna sau an).

Pentru exemplificarea funcției financiare PV, furnizăm următoarea aplicație în figura 1.82.

O persoană fizică dorește încheierea unei polițe de asigurare pentru o perioadă de 20 de ani cu o rată anuală a dobânzii de 40%. Asiguratul urmează să plătească lunar o primă de asigurare de 600.000 lei. Costul anuității perceput de asigurator este estimat la 19.000.000 lei. În figura următoare s-a calculat valoarea prezentă cu ajutorul funcției PV.

	A	B	
31	Rata anuală a dobânzii	40%	
32	Numărul de ani pe care se contractează asigurarea	20	
33	Valoarea lunară a plății primei de asigurare	600.000 LEI	
34	Valoarea prezentă este	-17.993.120 LEI	
35		=PV(B31/12;B32*12;B33)	

Fig. 1.82 Aplicație pentru funcții financiare

Din calculul făcut (în celula B34), reiese că valoarea prezentă a anuității este de 17.993.120 lei, adică mai mică decât valoarea anuității calculate de asigurator care este de 19.000.000 lei. Deci această investiție nu este rentabilă.

Se observă că rata dobânzii a fost exprimată în luni (rata anuală a fost împărțită la 12), iar numărul de ani pentru care s-a contractat asigurarea a fost exprimat tot în luni (numărul de ani a fost înmulțit cu 12).

De asemenea se observă că funcția PV a returnat un număr negativ. Explicația acestui rezultat este legată de faptul că funcția PV semnifică o cheltuială, o ieșire de bani. Pentru ca funcția să returneze un rezultat pozitiv, ar fi trebuit ca argumentul “valoarea plății” să fie introdus ca număr negativ (de exemplu –600.000).

=FV(rata dobânzii;numar de perioade[;marimea platii; valoarea prezenta; tipul]) returneaza valoarea viitoare (future value) a unei investitii sau plasament in conditii de anuitate (plati si rate ale dobânzii constante). Argumentele functiei financiare FV sunt identice ce cele ale functiei PV cu exceptia faptului ca unul din argumente reprezinta valoarea prezenta a investitiei sau plasamentului.

Pentru exemplificarea functiei financiare FV, oferim urmatoarea aplicatie: o persoana fizica doreste efectuarea unui plasament de 25.000.000 lei pe o perioada de 9 luni la o banca comerciala, pentru o dobânda anuala de 55%. Persoana fizica urmeaza a depune lunar la banca, alaturi de depozitul initial câte 1.000.000 lei pe aceeasi perioada.

In figura 1.83 s-a calculat (in celula B42) valoarea viitoare a sumei depuse la banca de respectiva persoana fizica (48.259037,70 lei).

	A	B	
38	Rata anuală a dobânzii	55%	
39	Durata plasamentului (luni)	9	
40	Valoarea inițială (prezentă) a plasamentului	25.000.000 LEI	
41	Valoarea plății în fiecare perioadă	1.000.000 LEI	
42	Valoarea viitoare este	48.259.037,70 LEI	
43		=FV(B38/12;B39;-B41;-B40)	

Fig. 1.83 Aplicatie pentru functii financiare (FV)

=PMT(rata dobânzii;numar de perioade;valoare prezenta[;valoare viitoare;tip]) calculeaza valoarea lunara sau anuala a platii pentru o investitie sau un imprumut.

Pentru exemplificarea functiei PMT (paiement), presupunem un imprumut la o banca comerciala pentru achizitionarea unui bun de folosinta indelungata in valoare de 35.000.000 lei. Rata dobânzii pentru creditele de consum este de 43% pe an, iar durata imprumutului a fost stabilita la 5 ani.

	A	B	C
46	Rata anuală a dobânzii	43%	
47	Durata împrumutului (ani)	5	
48	Suma împrumutată	35.000.000 LEI	
49	Valoarea lunară a plății	1.426.730 LEI	=PMT(B46/12;B47*12;-B48)
50	Costul total al împrumutului	85.603.809 LEI	=B49*B47*12
51	Valoarea totală a dobânzii	50.603.809 LEI	=B50-B48

Fig. 1.84 Aplicatie pentru functii financiare (PMT)

Valoarea lunara a platii catre banca pentru creditul acordat a fost calculata (in celula B49) prin functia PMT in figura 1.84:

De asemenea s-a calculat costul total al imprumutului ca un produs intre valoarea lunara a platii si numarul de perioade de plata in luni. Valoarea totala a dobânzii s-a calculat ca diferenta intre costul total al imprumutului si suma imprumutata.

=RATE(numar de perioade;valoarea platii;valoare prezenta) returneaza rata dobânzii pe perioada unei anuitati, pentru un imprumut sau o investitie. Pentru exemplificare presupunem efectuarea unui imprumut printr-un credit de 15 milioane lei pe timp de un an, cu o valoare lunara de rambursat in suma de 1.800.000 lei.

In figura 1.85 se calculeaza in celula E58, dobânda lunara perceputa de banca, iar in celula E59 dobânda anuala pentru suma imprumutata.

	A	B	C
55	Valoare împrumut	15.000.000 LEI	
56	Durata împrumutului (ani)	1	
57	Valoarea lunară a plății	1.800.000 LEI	
58	Dobânda lunară este	6,11 %	=RATE(B56*12;-B57;B55)
59	Dobânda anuală este	73 %	=B58*13

Fig. 1.85 Aplicație pentru funcții financiare (RATE)

Funcțiile financiare PMT, RATE și PV prezentate anterior, permit *construirea tablourilor de rambursare pentru împrumuturi*, aplicație exemplificată în figura 1.86.

Un întreprinzător particular solicită unei bănci comerciale un împrumut de 15 milioane lei pentru achiziționarea unui utilaj. Banca acordă creditul pe o durată de 6 luni cu o dobândă de 60% pe an.

Solicitantul creditului își poate întocmi în Excel un tablou de rambursare al împrumutului, altfel spus un scadențar.

	A	B	C	D
62	Suma împrumutată	15.000.000 LEI	Durata împrumutului (luni)	6
63	Dobânda anuală	60%		
64	Dobânda lunară	5,00%	Valoare lunară de rambursare	2.955.262,02 LEI
65				
66	Număr luni rămase	Rest de rambursat	Dobânda lunară	Rată de plată
67	6	15.000.000,00 LEI	750.000,00 LEI	2.205.262,02 LEI
68	5	12.794.737,98 LEI	639.736,90 LEI	2.315.525,12 LEI
69	4	10.479.212,86 LEI	523.960,64 LEI	2.431.301,38 LEI
70	3	8.047.911,48 LEI	402.395,57 LEI	2.552.866,45 LEI
71	2	5.495.045,03 LEI	274.752,25 LEI	2.680.509,77 LEI
72	1	2.814.535,26 LEI	140.726,76 LEI	2.814.535,26 LEI
73	0	0,00 LEI	#NUM!	#NUM!
74				
75	Numărul lunii pt rambursarea anticipată	3		
76	Suma de rambursat cu anticipație	8.047.911,48 LEI		

Fig. 1.86 Tabloul de rambursare a creditelor

Într-o primă fază, se poate calcula valoarea lunară de rambursare (celula D64) cu ajutorul funcției PMT.

În a doua fază, se organizează tabloul de rambursare pe patru coloane:

- prima coloană (coloana A) conține numărul lunii pentru care se face calculul dobânzii și restul de rambursat;

- a doua coloană (coloana B) conține restul de rambursat, adică suma care rămâne de restituit băncii la sfârșitul lunii în curs. Pentru prima lună restul de rambursat este egal cu creditul, adică 15 milioane, iar lunile următoare se calculează ca diferență între restul de rambursat și rata de plată;

- a treia coloană (coloana C) conține dobânda lunară calculată cu funcția RATE, adică suma lunară ce reprezintă dobânda calculată asupra restului de rambursat;

- a patra coloană (coloana D) reprezintă rata de rambursat, calculată ca diferență între valoarea lunară de rambursare (celula D64) și dobânda lunară.

Împrumutul este complet rambursat la începutul celei de-a șaptea perioadă, după ce a fost achitată a șasea rată.

Dacă s-ar fi dorit rambursarea integrală a împrumutului în luna a 4-a, deci când ar mai fi 3 luni de platit, suma de rambursat ar fi de 8.047.911,48 lei (suma a fost calculată cu funcția financiară PV).

Formulele utilizate pentru studiul de caz prezentat se găsesc ilustrate în figura 1.87.

	A	B	C	D
62	Suma împrumutată	15.000.000 LEI	Durata împrumutului (luni)	6
63	Dobânda anuală	60%		
64	Dobânda lunară	=B63/12	Valoare lunară de rambursare	=PMT(B64;D62;-B62)
65				
66	Număr luni rămase	Rest de rambursat	Dobânda lunară	Rată de plată
67	6	=B62	=RATE(A67;-\$D\$64;B67)*B67	=\$D\$64-C67
68	5	=B67-D67	=RATE(A68;-\$D\$64;B68)*B68	=\$D\$64-C68
69	4	=B68-D68	=RATE(A69;-\$D\$64;B69)*B69	=\$D\$64-C69
70	3	=B69-D69	=RATE(A70;-\$D\$64;B70)*B70	=\$D\$64-C70
71	2	=B70-D70	=RATE(A71;-\$D\$64;B71)*B71	=\$D\$64-C71
72	1	=B71-D71	=RATE(A72;-\$D\$64;B72)*B72	=\$D\$64-C72
73	0	=B72-D72	=RATE(A73;-\$D\$64;B73)*B73	=\$D\$64-C73
74				
75	Numărul lunii pt rambursarea	3		
76	Suma de rambursat cu anticipație	=PV(B64;B75;-D64)		

Fig. 1.87 Formulele tabloului de rambursare a creditelor

=NPER(rata dobânzii;valoarea platii;valoare prezenta) returneaza numarul de perioade de plata pentru o investitie sau un plasament. Altfel spus, se calculeaza câte varsaminte sunt necesare pentru ca un capital constituit printr-o investitie si remunerat printr-o dobânda sa atinga o valoare specificata.

	A	B	C
97	Rată lunară de rambursat	1.200.000 LEI	
98	Dobândă anuală	20%	
99	Dobândă lunară	1,67%	
100	Sumă împrumutată	35.000.000 LEI	
101	Număr perioade (ani)	=NPER(B98;-B97*12;B100)	

Fig. 1.88 Aplicatie pentru functii financiare (NPER)

In exemplul ilustrat in figura 1.88 s-a calculat numarul de perioade in ani in care un intreprinzator trebuie sa restituie un imprumut de 35 milioane lei, cu o dobânda anuală de 20%, platind lunar 1.200.000 lei.

=NPV(rata dobânzii;valoare1,valoarea2,.....) calculeaza valoarea actuala neta a unei investitii bazate pe o serie periodica de intrari de numerar (cash flows). Functia NPV difera de functia PV (present value), pentru ca se bazeaza pe varsaminte care nu au aceeasi marime. Astfel se calculeaza valoarea actuala neta a unor intrari viitoare de fonduri, pentru a se evalua rentabilitatea unei investitii. Intrarile de fonduri sunt operationale la intervale regulate, la sfârșitul fiecărei perioade.

Pentru exemplificarea functiei financiare NPV, furnizam urmatoarea aplicatie: o intreprindere doreste realizarea unei investitii de 170 milioane lei, care ii va permite intrarea unor fonduri estimate ca variabile pe parcursul a 6 ani. Aceste intrari de fonduri se presupun a fi de 223 milioane lei. Astfel se va pune problema rentabilitatii investitiei.

In figura 1.89 s-a construit un model economic, cu ajutorul caruia s-a calculat prin functia NPV (in celula B91) valoarea actuala neta a investitiei, care a fost de 95.291.904 lei daca rata dobânzii a fost de 25% (celula B80). A face o investitie de 170 milioane lei, antreneaza cheltuieli suplimentare de 74.708.096 lei fata de cheltuielile initial prevazute.

=SLN(valoare de inventar;valoare reziduala;durata normata de functionare) calculeaza amortismentul linear al unei imobilizari cu o valoare de inventar data, tinând cont de o valoare reziduala estimata, pentru un numar de perioade cât se presupune ca va functiona investitia.

Amortizarea lineara a unei imobilizari se face prin anuitati constante (anuitatea se calculeaza raportând valoarea de inventar a imobilizarii la durata de functionare a acesteia).

Toate functiile financiare pentru calculul amortismentului fac apel la notiunea de valoare reziduala. Aceasta notiune semnifica valoarea ce va putea fi recuperata la revânzarea imobilizarii.

	A	B	C	D	E
79	Valoarea investiției	170.000.000 LEI			
80	Rata anuală a dobânzii	25%			
81					
82		Intrări de numerar			
83	Anul 1	18.000.000 LEI			
84	Anul 2	24.000.000 LEI			
85	Anul 3	35.000.000 LEI			
86	Anul 4	42.000.000 LEI			
87	Anul 5	48.000.000 LEI			
88	Anul 6	56.000.000 LEI			
89	Total	223.000.000 LEI			
90					
91	Valoarea actuală netă	95.291.904,00 LEI	=NPV(B80;B83;B84;B85;B86;B87;B88)		
92	Diferență	-74.708.096,00 LEI	=B91-B79		

Fig. 1.89 Aplicatie pentru functii financiare (NPV)

Prin functia financiara SLN, anuitatile amortizarii lineare se calculeaza raportând diferenta dintre valoarea de inventar si valoarea reziduala la numarul de perioade cât a fost estimata durata de functionare a imobilizarii.

=VDB(valoare de inventar;valoare reziduala;durata normata de functionare;debutul perioadei;sfârșitul perioadei[rata de depreciere;comutator]) calculeaza amortismentul degresiv ajustat (variable declining balance) al unei imobilizari cu o valoare de inventar anume, o oarecare valoare reziduala; amortizabila pe mai multi ani; cu o anumita rata de depreciere.

Amortismentul degresiv ajustat reprezinta amortismentul contabil descrescator (amortismentul este mai mare pentru primele anuitati) pâna ce anuitatea amortismentului este mai mica decât anuitatea ce corespunde amortismentului linear, iar de aici incolo suma amortizabila este calculata linear.

Argumentele “debutul” si “sfârșitul perioadei” sunt utilizate pentru calcularea anuitatilor incomplete de amortisment, adica plecând de la o perioada când se incepe calculul amortizarii, catre ultima perioada pentru care se calculeaza amortizarea.

Rata de depreciere este un parametru care influenteaza amortizarea in sensul cresterii gradului de depreciere al imobilizarii.

Argumentul facultativ “comutator” permite sau trecerea automata de la amortizarea degresiva la amortizarea lineara (valoare zero – implicita) sau impiedicarea acestei treceri (valoare unu).

Pentru exemplificare, furnizam urmatoarea aplicatie: o societate comerciala efectueaza la inceputul anului o investitie de 10 milioane de lei, amortizabila in cinci ani. Rata de depreciere pentru o astfel de investitie cu o durata de functionare de 5 ani este de 2. Pentru ca investitia a fost finalizata la inceputul anului, aceasta este complet amortizata la sfârșitul celui de-al cincilea an.

Tabloul de amortizare aferent acestei imobilizari este prezentat in figura 1.90.

Pentru fiecare an, perioadele de debut si sfârșit sunt luate in calcul in coloanele C si D. Primul an de amortizare a imobilizarii incepe la 1 ianuarie (valoare 0) si dureaza pâna la 31 decembrie (valoare 1) si asa mai departe pentru anii urmatoari (intervalul fiind de 1).

B112		=	=VDB(\$B\$105,\$B\$106,\$B\$108;C112;D112;\$B\$107;\$B\$109)		
	A	B	C	D	
105	Valoare de inventar	10.000.000 LEI			
106	Valoare reziduală	0			
107	Rata de depreciere	2			
108	Durata normală de utilizare	5 ani			
109	Comutator	0			
110					
111	Ani	Amortizare	Perioadă debut	Perioadă sfârșit	
112	Anul 1	4.000.000 LEI	0	1	
113	Anul 2	2.400.000 LEI	1	2	
114	Anul 3	1.440.000 LEI	2	3	
115	Anul 4	1.080.000 LEI	3	4	
116	Anul 5	1.080.000 LEI	4	5	

Fig. 1.90 Tablou de amortizare (I)

Se remarca faptul ca ultimele doua anuitati sunt egale, metoda de calcul a amortizatii trecând automat de la procedeul degresiv la cel linear in anul patru.

Daca investitia ar fi fost pusa in functiune la mijlocul anului, perioada de debut – sfârșit aferenta primului an, ar fi fost 0 si 0,5 (celulele C112 si C113), iar daca investitia ar fi fost terminata la 1 aprilie, intervalul ar fi fost 0 si 0,75 (9/12 dintr-un an).

Daca tabloul de amortizare ar fi recalculat utilizând comutatorul 1, calculul degresiv ar fi fost complet, ultima transa de amortizare pentru anul 5 facându-se prin diferenta, dupa cum se observa si din figura 1.91.

B116		=	=VDB(\$B\$105,\$B\$106,\$B\$108;C116;D116;\$B\$107;\$B\$109)			
	A	B	C	D	E	
105	Valoare de inventar	10.000.000 LEI				
106	Valoare reziduală	0				
107	Rata de depreciere	2				
108	Durata normală de utilizare	5 ani				
109	Comutator	1				
110						
111	Ani	Amortizare	Perioadă debut	Perioadă sfârșit		
112	Anul 1	4.000.000 LEI	0	1		
113	Anul 2	2.400.000 LEI	1	2		
114	Anul 3	1.440.000 LEI	2	3		
115	Anul 4	864.000 LEI	3	4		
116	Anul 5	518.400 LEI	4	5		

Fig. 1.91 Tablou de amortizare (II)

=SYD(valoare de inventar;valoare reziduala;durata normata de functionare; perioada pentru care se calculeaza amortizarea) returneaza amortismentul degresiv absolut, fara a corecta ultimele anuitati pentru amortizarea completa a investitiei.

	A	B	C	D	E	F
3	Valoare inventar	80.000.000				
4	Valoare reziduală	10.000.000				
5	Durata	10			=DDB(\$B\$3;\$B\$4;\$B\$5;A10;3)	
6	Luni pe an	10				
7						
8	TIPURI DE AMORTIZARI					
9	Perioada	SLN Amortizare lineară	SYD Amortizare degresivă	DB Amortizare degresivă	DDB Amortizare degresiva 2	DDB Amortizare degresiva 3
10	1	70.000.000	12.727.273	12.533.333	16.000.000	24.000.000
11	2	35.000.000	11.454.545	12.683.733	12.800.000	16.800.000
12	3	23.333.333	10.181.818	10.299.191	10.240.000	11.760.000
13	4	=SLN(\$B\$3;\$B\$4;A13)		8.362.943	8.192.000	8.232.000
14	5	14.000.000	=SYD(\$B\$3;\$B\$4;\$B\$5;A14)		6.553.600	5.762.400
15	6	11.666.667	6.363.636	=DB(\$B\$3;\$B\$4;\$B\$5;A15;\$B\$6)		3.445.600
16	7	10.000.000	5.090.909	4.477.414	=DDB(\$B\$3;\$B\$4;\$B\$5;A16;2)	
17	8	8.750.000	3.818.182	3.635.660	3.355.443	0

Fig. 192 Aplicatii pentru functii economice

=DB(valoare de inventar;valoare reziduala;durata normata de functionare; perioada pentru care se calculeaza amortizarea;numarul de luni pe an de functionare a imobilizarii) returneaza amortismentul degresiv absolut, tinând cont de numarul de luni pe an de functionare a imobilizarii, deci ia in calcul un posibil aspect sezonier de utilizare al acesteia. Cele doua functii SYD si DB sunt calculate pentru o rata de depreciere egala cu 2.

=DDB calculeaza amortizarea dupa metoda softy si este o functie asemanatoare cu DB, cu exceptia faptului ca ultimul argument este un factor de multiplicare al amortizarii degresive. Astfel, anuitatile sunt calculate de o asa maniera incât ultima anuitate este n, penultima anuitate este 2n, antepenultima este 3n si asa mai departe.

Prezentam in figura 1.92 un exemplu edificator de utilizare a functiilor SLN, SYD, DB si DDB.

1.4.2 FUNCTII DEFINITE DE UTILIZATOR^[1] (categoria *User Defined*)

Funcțiile proprii sunt funcții definite de utilizator și care se comportă în esență ca orice funcție predefinită. Numele acestor funcții, după ce au fost definite, apar în caseta corespunzătoare categoriei **User Defined**, categorie creată în momentul definirii primei funcții utilizator.

Aceste funcții, odată definite, devin disponibile inclusiv prin asistentul de funcții, dar pot fi introduse și prin tastare directă în bara de formule.

Utilizatorul recurge la definirea de funcții proprii atunci când expresia de calcul este prea lungă și trebuie să o utilizeze frecvent (deci preferă o funcție care să abrevieze expresia de calcul respectivă) sau când conține calcule ce nu sunt posibil de efectuat doar cu ajutorul operatorilor utilizabili în formule.

Definirea unei funcții proprii se realizează cu ajutorul limbajului Visual Basic. Faptul că în lucrarea de față nu este prezentat acest limbaj, nu ne împiedică să arătăm modul în care se poate defini o funcție. Se procedează astfel:

- se alege comanda **Tools; Macro; Visual Basic Editor ; Insert Module**. În registrul de lucru activ este inserată o foaie al cărei nume implicit este **Module**. Este o foaie de lucru Visual Basic și diferă de foile de calcul atât prin structură cât și prin comenzile din bara de meniu. Foaia este inserată și deschisă.

- se poziționează cursorul în foaie și se tastează cuvântul **Function** urmat după un spațiu de numele funcției și de lista parametrilor plasată între paranteze;

- începând cu linia următoare se tastează instrucțiunile necesare pentru efectuarea prelucrărilor atribuite funcției;

- ultima linie din definirea funcției trebuie să conțină doar cuvintele obligatorii **End Function** . Aici se încheie procesul de definire a funcției.

Pentru înțelegerea procesului de definire a unei funcții proprii, furnizăm următorul exemplu: să se definească o funcție numită *Spor*, care pe baza salariului și a vechimii unui angajat, să calculeze sporul de vechime ce i se cuvine. Se știe că algoritmul de calculare a sporului de vechime este următorul:

- pentru o vechime sub 3 ani nu se acorda spor;
- pentru o vechime între 3 și 5 ani sporul reprezintă 5% din salariu;
- pentru o vechime între 5 și 10 ani sporul este de 10% din salariu;
- pentru o vechime între 10 și 15 ani sporul este 15% din salariu;
- pentru o vechime > 15 ani sporul este 20% din salariu.

Urmând procedeul prezentat anterior, vom obține foaia **Module** (pe care am redenumit-o sugestiv **Funcții proprii**) figura 1.93.

^[1] se mai numesc și funcții proprii

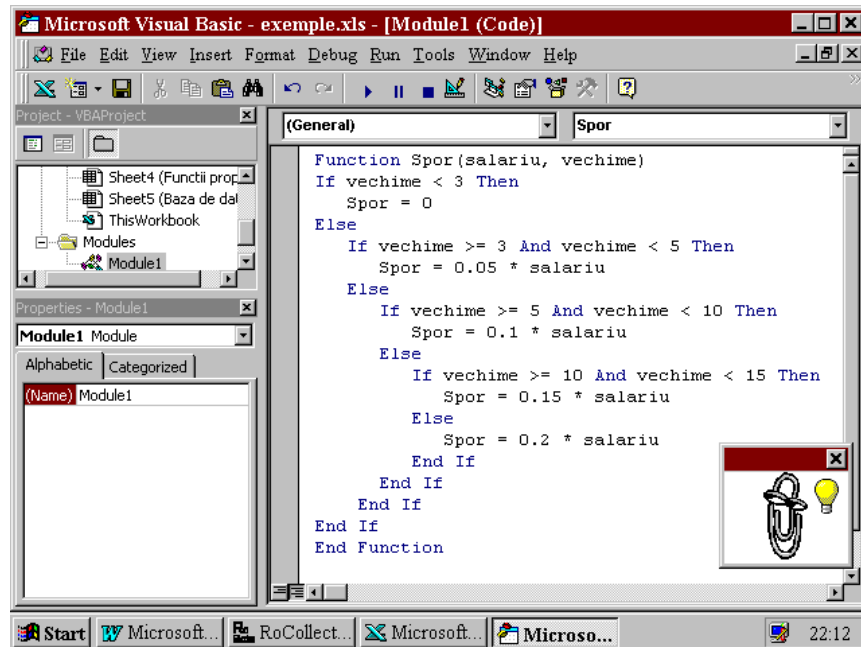


Fig. 193 Aplicatie functii proprii

Se procedeaza ca pentru orice alta functie predefinita. Pentru exemplificare, vom utiliza functia **Spor** pentru a calcula sporul de vechime aferent angajatilor oficiului de calcul.

Sursa de date este prezentata in figura 1.94.

Pentru rezolvarea aplicatiei, se parcurg urmatoorii pasi:

	A	B	C	D	E	F	G	H	I
1	Marca	Nume	Prenume	Functie	Vechime	Salariu	Spor		
2	E03	Dokroi	Ion	sef oficiu	15	180000			
3	103	AnLu	vlad	programator	5	80000			
4	E01	Aldea	Dumitr	inginer	3	120000			
5	E05	Marin	Cleuda	abzant	0	40000			
6	114	Misa	Ion	programator	1	111111			
7	194	Misa	Valer	abzant	4	411111			
8	702	Barcu	Catalina	abzant	10	40000			
9	703	Adem	Bogdan	analist	8	120000			
10	E02	Sabin	Sorna	analist	8	100000			
11	E07	Misa	Toms	deserator	2	60000			
12	E05	Misa	Gheorghe	abzant	1	70000			
13	E03	Barcu	Rauca	operator	2	50000			

Fig. 194 Sursa de date pentru aplicatie

1. se deschide foaia de calcul care contine lista angajatilor cu toate informatiile aferente (marca, nume, prenume, vechime, salariu);
2. se adauga listei o coloana intitulata spor vechime;
3. in celula corespunzatoare primului angajat se introduce formula de calcul printr-una din metodele cunoscute:
 - a) se tasteaza: =SPOR(F2;E2) unde F2, E2 sunt coordonatele corespunzatoare salariului si vechimii angajatului respectiv
 - b) se apeleaza asistentul de functii. In prima fereasta a **asistentului Paste Function** se selecteaza din categoria **User Defined**, functia **Spor** (figura 1.95a).

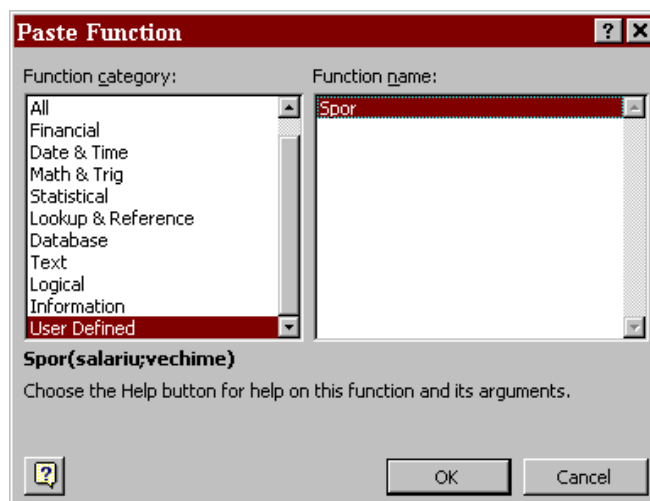


Fig. 195a Asistentul de functii

In a doua fereastra a asistentului de functii se precizeaza valorile parametrilor (fig.1.95b)

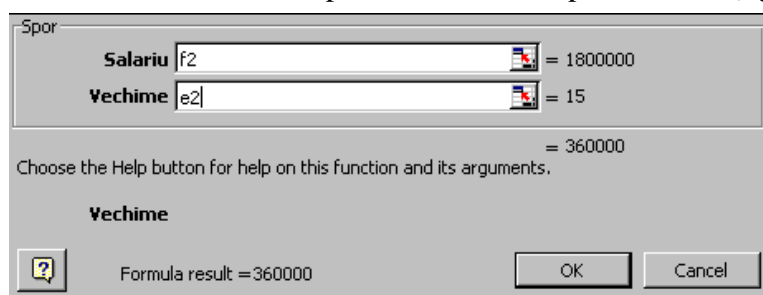


Fig. 195b Fereastra parametri

Prin validarea formulei, in celula este afisat rezultatul calculului. Pentru a calcula sporul de vechime pentru toti angajatii se copiaza formula. Rezultatul final este prezentat in figura 1.96.

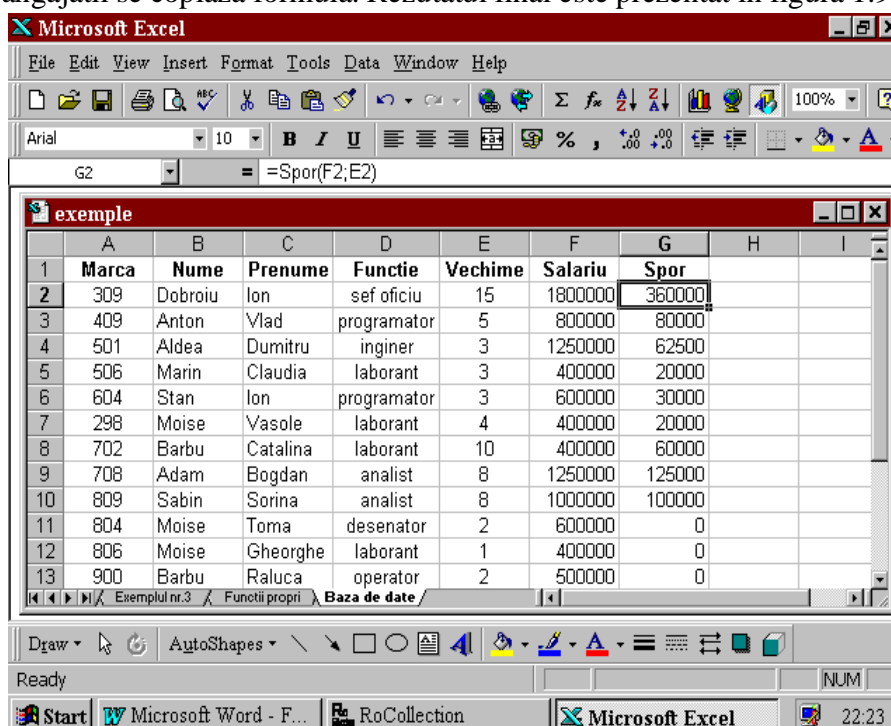


Fig. 196 Rezultatul aplicatiei cu functii proprii