1. Cantitatea de informaţie

Cantitatea de informaţie **I** ce se conţine într-un mesaj emis de sursă cu număr finit de mesaje se determină din relaţia: $\mathbf{I=log_a n}$, unde n este numărul de mesaje posibile ale sursei. Dacă unitatea de măsură a cantităţii de informaţie este bitul (cantitatea de informaţie într-un mesaj al unei surse cu numai 2 mesaje posibile) atunci $\mathbf{I=log_2 n}$.
Dacă se cunoaşte cantitatea de informaţie I ce se conţine într-un mesaj, cantitatea totală de informaţie emisă de sursă se determină din relaţia: $\mathbf{V=N*I=N*log_2 n}$, unde N este numărul de mesaje transmise.

Tipuri de probleme:

1. Cantitatea de informaţie în 1000 de mesaje emise de o sursă este egală cu 4000 biţi. Calcutaţi numărul de mesaje posibile al acestei surse.

Este dat:	Rezolvare:
V=4000 biţi	
N=1000	$V=N*I=N*\log_2 n$ => $\log_2 n=V/N=4000/1000=4$ => $n=2^4=16$ (mesaje posibile)
n-?	

2. Cantitatea de informaţie emisă de o sursă cu 8 mesaje posibile este egală cu 3000 biţi. Calcutaţi numărul de mesaje emise de această sursă.

Este dat:	Rezolvare:
V=3000 biţi	
n=8	$V=N*I=N*\log_2 n$ => $N=V/\log_2 n=3000/\log_2 8=3000/3$ => N=1000 (mesaje emise)
N-?	

3. Cantitatea de informaţie în informaţiile textuale.

Pe o pagină pot fi tipărite 60 de rînduri de text. Fiecare rînd conţine 90 de caractere. Cîte pagini pot fi memorate în formă nearhivată pe o dischetă de 1,44 MB?

Rezolvare:
Codificarea textului se realizează prin înlocuirea fiecărui caracter cu un cuvînt binar 8-poziţional, adică, cu un octet (Byte). Numărul de caractere de pe o pagină se estimează ca produsul dintre numărul de rînduri şi numărul de caractere în fiecare rînd: 90*60=5400.
De aici, cantitatea de informaţie ce se conţine într-o pagină de text este 5400 B.
Numărul de pagini ce pot fi memorate în formă nearhivată pe o dischetă de 1,44 MB este egal cu $1,44*10^6/5400\approx167$ (pagini)

2. Cuantizarea imaginilor

Pentru a evalua cantitatea de informaţie, imaginea este împărţită în microzone cu ajutorul unui rastru. Fiecare microzonă se descrie prin luminanţa sa. Această mărime poate fi discretizată în valoare (cuantificată). Numărul cuantelor n caracterizează puterea de rezoluţie a echipamentelor pentru formarea imaginilor. Cantitatea de informaţie a unei imagini monocrome: $\mathbf{I=m_x m_y log_2 n}$, unde $\mathbf{m_x}$ şi $\mathbf{m_y}$ reprezintă numărul de microzone ale rastrului respectiv pe orizontală şi verticală. Întrucît culorile pot fi redate prin suprapunerea a trei reprezentări ale aceleiaşi imagini în roşu, verde şi albastru, cantitatea de informaţie dintr-o imagine color se determină din relaţia: $\mathbf{I=3m_x m_y log_2 n}$.

1. Evaluaţi cantitatea de informaţie într-o fotografie monocrom (sau color) cu dimensiunile 10x10 cm redată cu ajutorul unui rastru cu rezoluţia 24 puncte/cm. Fiecare punct poate avea 256 nivele de luminanţă.

Rezolvare:
Numărul de puncte $m_x=10*24=240$;
Numărul de puncte $m_y=10*24=240$;

Cantitatea de infornaţie ce se conţine în fiecare punct este $\log_2 256=8$ (biţi);
Cantitatea de informaţie într-o fotografie monocrom **$I=m_x m_y \log_2 n=240*240*8=460800$ (biţi)**
Cantitatea de informaţie într-o fotografie color **$I=3*m_x m_y \log_2 n=240*240*8=4147200$ (biţi)**

2. Cîte nivele de luminanţă pot fi redate pe ecran dacă cuvintele imaginii numerice sînt 4-poziţionale?

Rezolvare:
Cuvintele imaginii numerice sînt 4-poziţionale, de aici rezultă că cantitatea de infornaţie ce se conţine în fiecare punct este $I=\log_2 n=4$ (biţi).
Din această egalitate găsim numărul de nivele de luminanţă ce pot fi redate pe ecran: **$n=2^4=16$** (nivele de luminanţă).

3. Codificarea şi decodificarea informaţiei

Se numeşte **semn** un element al unei mulţimi finite de obiecte ce se pot distinge. O mulţime de semne ordonate liniar se numşte **alfabet**.
Un şir finit de m semne, dintre care unele se pot repeta , formează un cuvînt.
Cuvintele formate din semne binare se numesc **cuvinte binare**. Dacă lungimea cuvintelor binare este constantă, ele se numesc **m-poziţionale**.
Regula de transformare a mesajelor în cuvinte se numeşte **cod**, iar operaţia respectivă – **codificare**. Operaţia inversă codificării se numeşte **decodificare**.
Cel mai simplu este codul în care mesajelor posibile s_1, s_2, …s_n le corespund cuvinte binare de lungime constantă m (cod m-poziţional). Decodificarea va fi univocă numai dacă lungimea m a cuvîntului de cod satisface inegalitatea **$2^m \geq n$**. Lungimea cuvintelor unui cod poziţional trebuie să fie mai mare sau egală cu cantitatea de informaţie a unui mesaj: **$m \geq \log_2 n$**.

Tipuri de probleme.

1. *Cod Bacon – codificare, decodificare.*

În codul propus de filozoful englez Francis Bacon literele alfabetului latin se prezintă astfel:

A — 00000 B — 00001 C — 00010 D — 00011..... Z — 11001.

Decodificaţi, utilizînd codul Bacon , următorul şir binar:

01000 00011 00100 00000.

Rezolvare:

Construim tabelul de codificare şi găsim literele respective secvenţelor binare din condiţia problemei: **IDEA.**

A	**00000**	N	01101
B	00001	O	01110
C	00010	P	01111
D	**00011**	Q	10000
E	**00100**	R	10001
F	00101	S	10010
G	00110	T	10011
H	00111	U	10100
I	**01000**	V	10101
J	01001	W	10110
K	01010	X	10111
L	01011	Y	11000
M	01100	Z	11001

2. *Reprezentarea în cod poziţional a indicaţiilor numerice de forma zz.ll.aa (zz-ziua, ll-luna, aa-anul, ultimele 2 cifre) a unui calendar electronic este 0101101010000010. Decodificaţi acest mesaj. Motivaţi răspunsul.*

Rezolvare:
Numărul mesajelor posibile ce se conţin în zz este 31 (S_{zz}={1..31}). Cantitatea de informaţie într-un mesaj $I_{zz}=\log_2 31$. Lungimea minimă a cuvintelor m-poziţionale se determină din inegalitatea: $m_{zz} \geq \log_2 31=5$ (biţi).
Numărul mesajelor posibile ce se conţin în *ll* este 12 (S_{ll}={1..12}). Cantitatea de informaţie într-un mesaj $I_{ll}=\log_2 12$. Lungimea minimă a cuvintelor m-poziţionale se determină din inegalitatea: $m_{ll} \geq \log_2 12=4$(biţi).
Numărul mesajelor posibile ce se conţin în *aa* este 100 (S_{aa}={00..99}). Cantitatea de informaţie într-un mesaj $I_{zz}=\log_2 100$. Lungimea minimă a cuvintelor m-poziţionale se determină din inegalitatea: $m_{aa} \geq \log_2 100=7$ (biţi).
Reprezentarea în cod poziţional a indicaţiilor numerice de forma zz.ll.aa (zz-ziua, ll-luna, aa-anul, ultimele 2 cifre) a unui calendar electronic este formată din 3 cuvinte binare: Primul are 5 biţi şi reprezintă ziua: $(01011)_2=(11)_{10}$
Al doilea are 4 biţi şi reprezintă luna: $(0101)_2=(05)_{10}$
Al treilea are 7 biţi şi reprezintă ultimele două cifre ale anului: $(0000010)_2=(02)_{10}$.
Aşadar: **01011 0101 0000010** este reprezentarea binară a următoarei date calendaristice **11.05.02**

3. *Scrieţi lungimea minimă a şirurilor binare, necesară pentru codificarea şi decodificarea univocă a informaţiei, dacă mulţimea mesajelor posibile ale sursei de informaţie este **S={ 0 , 1 , … , 9, A, B, C, … , Z}**. Argumentaţi răspunsul.*
Rezolvare:
Lungimea cuvintelor m-poziţionale binare pentru codificarea mesajelor din această sursă, care are 10+26=36 mesaje se determină din inegalitatea: $m \geq \log_2 36$; m=6 (biţi)

1. *Care este numărul secvenţelor binare distincte formate din 3 biţi? Scrieţi aceste secvenţe.*
Rezolvare:
Fiecare poziţie binară poate conţine una din două valori posibile: 0 sau 1. Numărul total de secvenţe binare este $2^n=2^3=8$. Vom scrie aceste secvenţe:
000, 001, 010, 011, 100, 101, 110, 111

5. *Codificarea caracterelor unui text poare fi făcută utilizînd cifrele ternare 0, 1 şi 2. De exemplu, simbolului **A** îi va corespunde codul ternar (6-poziţional)* $\boldsymbol{(01000001)_2 = (65)_{10} =}$

(002102)₃

__(002102)__$_3$*, simbolului B îi va corespunde codul ternar* ***(01000010)***$_2$ ***= (66)***$_{10}$ ***= (002110)***$_3$ *ş. a. m. d. În astfel de cazuri unitatea de măsură a informaţiei este tritul.*

Cantitatea de informaţie dintr-un text este 240000 triţi. Exprimaţi această cantitate în biţi şi în octeţi.

Rezolvare:
Caculăm iniţial număru de caractere codificate cu 240000triţi: 240000(triţi)/6(triţi)=40000 (caractere). Fiecare caracter se codifică cu 8 biţi. De aici, aceeaşi cantitate de informaţie exprimată în biţi este 40000*8 (biţi)=320000 (biţi)=40000 (octeţi)

Sisteme de numeraţie

Totalitatea regulilor de reprezentare a numerelor, împreună cu mulţimea cifrelor poartă denumirea de **sistem de numeraţie** . Numărul cifrelor defineşte **baza** sitemului de numeraţie.
Sistemele în care semnificaţia cifrelor depinde de poziţia ocupată în cadrul numerelor se numesc **sisteme de numeraţie poziţionale**.
Presupunem că numărul N are partea întreagă formată din n+1 cifre, iar partea fracţionară – din m cifre:

$$N = c_n c_{n-1} \dots c_1 c_0, c_{-1} c_{-2} \dots c_{-m}$$

Valoarea acestui număr se evaluează în funcţie de baza sistemului:
$(N)_b = c_n b^n + c_{n-1} b^{n-1} + \dots + c_1 b^1 + c_0 b^0 + c_{-1} b^{-1} + \dots + c_{-m} b^{-m}$.
Conversiunea numărului zecimal $(N)_{10}$ în echivalentul său în baza b se efectuează conform următoarelor reguli:

- se împarte la baza respectivă partea întreagă şi cîturile obţinute după fiecare împărţire pînă se obţine cîtul zero: rezultatul conversiunii părţii întregi este constituit din resturile obţinute, considerate în ordinea inversă de apariţie;
- se înmulţeşte cu baza partea fracţionară, apoi toate părţile fracţionare obţinute din produsul anterior, pînă cînd partea fracţionară a unui produs este zero sau pînă la obţinerea unui număr de cifre fracţionare dorit; rezultatul conversiunii părţii fracţionare este constituit din părţile întregi ale produselor, considerate în ordinea apariţiei.

Conversiunea binar-octală sau octal-binară se face direct, reieşind din faptul că fiecare cifră octală se reprezintă prin 3 cifre binare.
Conversiunea binar-hexazecimală sau hexazecmal-binară se face direct, reieşind din faptul că fiecare cifră hexazecimală se reprezintă prin 4 cifre binare.

1. Transformaţi numărul ***(45,15)***$_{10}$ în sistemul binar şi sistemul octal de numeraţie.

Partea întreagă poate fi convertită în baza 2 aşa:

45:2=22 (1)
22:2=11 (0)
11:2=5 (1)
5:2=2 (1)
2:2=1 (0)
1:2=0 (1)

1 0 1 1 0 1

Fracţia se converteşte în baza 2 astfel:

0,15*2=0,3
0,3*2 =0,6
0,6*2 =1,2
0,2*2 =0,4
0,4*2=0,8
0,8*2=1,6

0 0 1 0 0 1

(45,15)$_{10}$ =$(101101,001001)_2$
Pentru a transforma numărul dat în baza 8 vom folosi conversiunea zecimal-octală.

Partea întreagă poate fi convertită în baza 8 aşa:

45:8=5 (5)
5:8=0 (5)

5 5

Fracţia se converteşte în baza 2 astfel:

0,15*2=0,3
0,3*2 =0,6
0,6*2 =1,2
0,2*2 =0,4
0,4*2=0,8
0,8*2=1,6

0 0 1 0 0 1 (1 0 0 1)

Pentru a transforma numărul dat în baza 8 putem folosi şi conversiunea binar-octală.
Vom grupa, începînd de la virgulă, a cîte 3 cifre binare.
Apoi vom converti fiecare grup de cifre binare în baza 8 (de exemplu: $101=1*2^2+0*2^1+1*2^0=5$).
Cifrele se scriu în ordinea lor de apariţie:
$(101\ 101,001\ 001\ 100\ 110)_2 = (55,1146)_8$

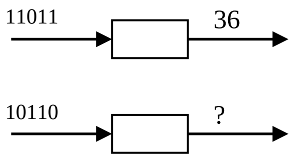
2. *Transformaţi numărul* ***(C,DB3)$_{16}$*** *în sistemele binar şi octal de numeraţie.*

Conversiunea hexazecimal-binară se realizează prin reprezenterea cifrelor hexazecimale componente ale numărului în baza 2 conţinînd fiecare a cîte 4 biţi (cantitatea de informaţie în 16 mesaje – cifrele hexazecimale - este: I= $\log_2 16=4$, de aici, lungimea codului binar m-poziţional este 4).

<table>
<tr><td>Cifra hexazecimală</td><td>Reprezentarea binară</td><td rowspan="17">(C,DB3)₁₆=(1100,1101 1011 0011)₂

Pentru a transforma numărul dat în baza 8 vom folosi conversiunea binar-octală. Vom grupa, începînd de la virgulă, a cîte 3 cifre binare.
Apoi vom converti fiecare grup de cifre binare în baza 8.
(C,DB3)₁₆= (001 100,110 110 110 011)₂=(14,663)₈</td></tr>
<tr><td>0</td><td>0000</td></tr>
<tr><td>1</td><td>0001</td></tr>
<tr><td>2</td><td>0010</td></tr>
<tr><td>3</td><td>0011</td></tr>
<tr><td>4</td><td>0100</td></tr>
<tr><td>5</td><td>0101</td></tr>
<tr><td>6</td><td>0110</td></tr>
<tr><td>7</td><td>0111</td></tr>
<tr><td>8</td><td>1000</td></tr>
<tr><td>9</td><td>1001</td></tr>
<tr><td>A</td><td>1010</td></tr>
<tr><td>B</td><td>1011</td></tr>
<tr><td>C</td><td>1100</td></tr>
<tr><td>D</td><td>1101</td></tr>
<tr><td>E</td><td>1110</td></tr>
<tr><td>F</td><td>1111</td></tr>
</table>

1. "Cutia neagră" transformă numerele scrise în baza 10 în altă bază. Determinaţi această bază şi numărul la ieşirea acesteia în cazul al doilea .



Rezolvare:
Reprezentăm în baza 10 primul număr: $(11011)_2=1*2^4+1*2^3+0*2^2+1*2^1+1*2^0=(27)_{10}$
$(27)_{10}=3*b^1+6*b^0$;
Rezolvăm ecuaţia 3b+6=27; 3b=21; b=7;
Reprezentăm numărul $(10110)_2=1*2^4+0*2^3+1*2^2+1*2^1+0*2^0=(22)_{10}$
Convertim acest număr în baza 7: 22:7=3 (1)
3:7=0(3)
$(31)_7$
Răspuns: baza în care se reprezintă numărul la ieşirea din "cutie neagră" este 7.
Numărul la ieşire este $(31)_7$

Reprezentarea numerelor în calculator

Reprezentarea numerelor întregi în calculator se realizeazî pe un număr fix de poziţii binare. Se cunosc trei moduri de reprezentare, numite coduri binare pentru numere algebrice.
Cod direct: o poziţie, prima din stînga, este rezervată semnului. Dacă în această poziţie este înscris 0, numărul binar este pozitiv, dacă 1 – numărul reprezentat este negativ.
Intervalul posibil admis de reprezentarea dată este $\mathbf{[-2^{n-1}+1,\ 2^{n-1}-1]}$
Cod invers: Pentru numerele pozitive scrierea în cod invers este identică cu cea din cod direct. Dacă numărul este negativ, el se înscie aşa cum ar fi pozitiv, apoi se inversează fiecare cifră binară. Pe n poziţii binare pot fi reprezentate numere întregi din intervalul
$\mathbf{[-2^{n-1}+1,\ 2^{n-1}-1]}$
Cod complementar: Scrierea numerelor pozitive în cod invers este identică cu cea din cod direct. Dacă numărul este negativ, el se înscie în cod invers, apoi se adună 1 la cifra cea mai puţin semnificativă. Pe n poziţii binare pot fi reprezentate numere întregi din intervalul $\mathbf{[-2^{n-1},\ 2^{n-1}-1]}$.

Numerele reale se reprezintă în calculator sub formă fracţionară cu virgulă fixă sau cu virgulă flotantă.
Reprezentarea în **virgulă fixă** se realizează considerînd că virgula este plasată imediat după poziţia cifrei semn. Virgula însăşi nu este materializată fizic.
Pe n poziţii binare pot fi reprezentate numere reale valoarea absolută a cărora este
$0,00\ldots0 \le |x| \le 0,11\ldots1$, sau în sistem zecimal, $0 \le |x| \le 1-2^{-(n-1)}$.
Numerele subunitare, ca şi numere întregi, pot fi reprezentate în cod direct, cod invers şi cod complementar.

1. *Codul direct a unui număr întreg reprezentat pe 8 biţi este:*

1	0	0	0	1	1	1	0

Scrieţi acest număr în sistemul zecimal de numeraţie.
Rezolvare:
În poziţia superioară este înscrisă valoarea 1, înseamnă că numărul reprezentat este negativ.
Convertim în sisemul zecimal reprezentarea binară: $1*2^3+1*2^2+1*2^1=14$.
Aşadar, numărul reprezentat pe 8 biţi este -14.

2. *Codul invers al unui număr întreg reprezentat pe **8 biţi** este:*

1	0	0	0	1	1	1	0

Scrieţi acest număr în sistemul zecimal de numeraţie.
Rezolvare:

Fiindcă numărul reprezentat în cod invers conţine în cifra semnului o unitate, numărul reprezentat este negativ. Pentru a scrie acest număr în sistemul zecimal:

a. Inversăm biţii:

0	1	1	1	0	0	0	1

b. convertim în baza 10: $1*2^0+0*2^1+0*2^2+0*2^3+1*2^4+1*2^5+1*2^6=113$
Numărul reprezentat este $(-113)_{10}$

2. *Codul complementar a unui număr întreg reprezentat pe 8 biţi este:*

1	0	0	1	0	1	0	0

Scrieţi acest număr în sistemul zecimal de numeraţie.

Fiindcă numărul reprezentat în cod complementar conţine în cifra semnului o unitate, numărul reprezentat este negativ. Pentru a scrie acest număr în sistemul zecimal:

a. Scădem o unitate din bitul inferior:

1	0	0	1	0	0	1	1

b. Inversăm biţii:

0	1	1	0	1	1	0	0

c. Convertim numărul pozitiv în baza 10:
$0*2^0+0*2^1+1*2^2+1*2^3+0*2^4+1*2^5+1*2^6=4+8+32+64=108$

Numărul reprezentat este $(-108)_{10}$.

4. *Prezentaţi în **virgula fixă** pe 8 poziţii binare în cod invers numărul negativ -0,0625*
În virgula fixă se reprezintă numerle subunitare, aşadar numărul dat nu necesetă modificări suplementare.
Convertim fracţia în sistemul binar de numerţie:
0,0625*2=**0**,125
0,125*2=**0**,25
0,25*2=**0**,5
0,5*2=**1**,0
$(0,0625)_{10}=(0,0001)_2$
Reprezentăm pe 8 biţi parea fracţionară a numărului pozitiv.
Bitul superior este rezervat semnului. Virgula zecimală nu se materializează. Partea fracţionară se reprezintă pe 7 biţi.

0	0	0	0	1	0	0	0

Pentru a reprezenta numărul în cod invers, inversăm biţii:

1	1	1	1	0	1	1	1

Algebra booleană

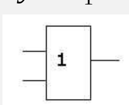
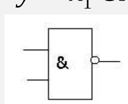
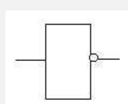
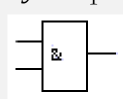
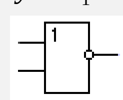
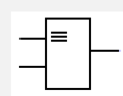
Algebra booleană poate fi definită printr-o mulţime a elementelor {0,1}, o mulţime a operatorilor elementari {-, &, V} (negaţia, conjuncţia, disjuncţia) şi printr-un număr de postulate. Orice variabilă a algebrei booleane poate avea numai una din două valori posibile, notate simbolic prin 0 şi 1. Operatorii elementari se definesc cu ajutorul tabelelor de adevăr. Tabelul de adevăr este un tabel care include toate combinaţiile posibile ale valorilor variabilelor faţă de care este definit operatorul şi rezultatul operaţiei respective.

X	$\bar{x}$
0	1
1	0

X	y	x&y
0	0	0
0	1	0
1	0	0
1	1	1

x	Y	$x \vee y$
0	0	0
0	1	1
1	0	1
1	1	1

Variabilele şi constantele logice, reunite cu ajutorul operatorilor logice formează expresii logice. Valorile expresiilor logice pot fi calculate cu ajutorul tabelelor de adevăr ale expresiilor logice, care includ toate combinaţiile posibile ale valorilor variabilelor din expresiia examenată şi rezultatele operaţiilor logice în ordinea calculării lor.
Pentru calcularea expresiilor logice este stabilită următoarea prioritate a operaţiilor logice: 1. negaţia, 2. conjuncţia, 3. disjuncţia.
Funcţia logică de n variabile $y = f(x_1, x_2, ..., x_n)$ este o aplicaţie care pune în corespondenţă fiecărei combinaţii de valori ale variabilelor $x_1, x_2, ..., x_n$ valoarea 0 sau 1 a variabilei y. Tabelul de adevăr al funcţiei logice $y = f(x_1, x_2, ..., x_n)$ este un tabel care include toate combinaţiile posibile ale valorilor argumentelor $x_1, x_2, ..., x_n$ şi valorile corespunzătoare ale variabilei dependente y.
Definirea funcţiei logice prin formule se face atribuind variabilei y vlorile expresiilor logice ce conţin argumentele $x_1, x_2, ..., x_n$.
Circuitul logic este un dispozitiv destinat calculării funcţiilor logice.
Circuitele destinate calculării funcţiilor logice frecvent utilizate se numesc circuite logice elementare sau porţi logice

$y = x_1 \vee x_2$	$y = \overline{x_1 \,\&\, x_2}$	$y = \bar{x}$	$y = x_1 \,\&\, x_2$	$y = \overline{x_1 \vee x_2}$	$y = x_1 x_2 \vee \bar{x}_1 \bar{x}_2$
					
SAU	SI-NU	NU	SI	SAU-NU	COINCIDENTA

Tipuri de probleme:

1. *Care din următoarele expresii logice sunt egale? (Două expresii logice sunt egale, dacă valorile lor coincid pentru toate combinaţiile posibile ale valorilor variabilelor din expresiile respective)*

Expresii

1. $(\bar{x} \vee \bar{y})(x \vee y)$
2. $x\bar{y} \vee \bar{x}y$
3. $xy \vee \overline{xy}$
4. $(\bar{x} \vee y)(x \vee \bar{y})$

Răspunsuri posibile

a) 1. şi 4.
b) 1. şi 2.
c) 2. şi 3.
d) 3. şi 4.

Rezolvare:
Completăm tabelul de adevăr pentru toate expresiile date.
Sunt egale acele expresii pentru care valorile ppentru toate combinările posibile coincid

X	Y	$(\bar{x} \vee \bar{y})(x \vee y)$	$x\bar{y} \vee \bar{x}y$	$xy \vee \overline{xy}$	$(\bar{x} \vee y)(x \vee \bar{y})$
0	0	0	0	1	1

0	1	1	1	0	0
1	0	1	1	0	0
1	1	0	0	1	1

Răspunsul corect este d)

2. Funcţia logică y este definită prin tabelul de adevăr:

X_1	X_2	Y
0	0	1
0	1	0
1	0	0
1	1	1

Scrieţi funcţia (funcţiile) din cele ce urmează, definite prin formulă, ce corespunde aceluiaşi tabel de adevăr.

a) $y = \bar{x}_1 x_2 \vee x_1 \bar{x}_2$

b) $y = (x_1 \vee \bar{x}_2)(\bar{x}_1 \vee x_2)$

c) $y = x_1 x_2 \vee \overline{x_1} x_2$

x_1	X_2	$y = \bar{x}_1 x_2 \vee x_1 \bar{x}_2$	$y = (x_1 \vee \bar{x}_2)(\bar{x}_1 \vee x_2)$	$y = x_1 x_2 \vee \overline{x_1} x_2$
0	0	0	1	0
0	1	1	0	1
1	0	1	0	0
1	1	0	1	1

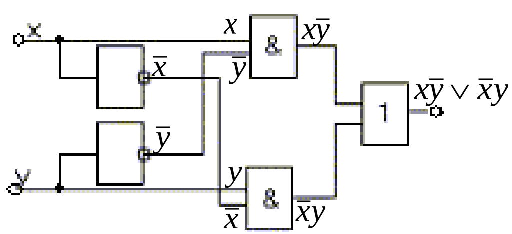
Răspunsul corect este b)

3. *Scrieţi funcţia, din cele de mai jos, materializată în acest circuit combinaţional.*

a) $f(x, y) = x\bar{y} \vee \bar{x}y$

b) $f(x, y) = (\bar{x} \vee \bar{y})(x \vee y)$

c) $f(x, y) = xy \vee \overline{xy}$



Răspunsul corect este a)

4. Se consideră următoarea funcţie logică: $y = x_1 \bar{x}_2 \vee x_2 x_3$.

Construiţi circuitul combinaţional, care realizează această funcţie.