Задаци за вежбу

1. ЗАДАТАК Превести неозначене целе бројеве $(16450)_7$ и $(B48)_{14}$ у основу 11 директним алгоритмом, а затим их и сабрати у тој основи.

2. ЗАДАТАК Бројеве $(+7)_{10}$ и $(-7)_{10}$ записати у бинарном потпуном комплементу помоћу 4 цифре, а затим их помножити произвољним алгоритмом за множење бројева у потпуном комплементу и добијени резултат пребацити у декадни систем.

3. ЗАДАТАК Број 45.24 записати према стандарду IEEE-754 у двострукој тачности. Такође, одредити декадне вредности следећих записа према IEEE-754 стандарду:
а) 0 10000110 10010001100000000000000
б) 0 11111111 00100011111110000001110
в) 1 00000000 00110001100000000000000

4. ЗАДАТАК Дат је текст "**Fred fed Ted bread, and Ted fed Fred bread**". Хафмановим кодирањем одредити оптималан префиксно слободан код за кодирање овог текста и израчунати колико битова је потребно за кодирање целог текста добијеним кодирањем.

5. ЗАДАТАК Методом цикличне провере редундантности испитати да ли је дошло до грешке при пријему поруке 10001110101, ако је полином генератор $G(x) = x^3 + 1$.

6. ЗАДАТАК Методом Карноове мапе минимизовати следећу логичку функцију:

a	b	c	d	$f(a,b,c,d)$
0	0	0	0	1
0	0	0	1	1
0	0	1	0	?
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

1. ЗАДАТАК

•$(16450)_7 \xrightarrow{?} (x)_{11}$.
У овом задатку до решења долазимо директним алгоритмом, односно не вршимо међупревођење у декадни бројевни систем.

Развој броја $(16450)_7$ је:
$1 \cdot 7^4 + 6 \cdot 7^3 + 4 \cdot 7^2 + 5 \cdot 7 + 0 = (((1 \cdot 7 + 6) \cdot 7 + 4) \cdot 7 + 5) \cdot 7 + 0$
Да бисмо добили тражени запис у основи 11, све операције у последњем изразу вршимо у основи 11.
Дакле:

$$(((1 \cdot 7 + 6) \cdot 7 + 4) \cdot 7 + 5) \cdot 7 + 0 = (((7 + 6) \cdot 7 + 4) \cdot 7 + 5) \cdot 7 + 0 = ((12 \cdot 7 + 4) \cdot 7 + 5) \cdot 7 + 0$$

$$= ((83 + 4) \cdot 7 + 5) \cdot 7 + 0 = (87 \cdot 7 + 5) \cdot 7 + 0 = (555 + 5) \cdot 7 + 0 = 55A \cdot 7 + 0 = 3584$$

Одакле добијамо да је запис броја $(16450)_7$ у основи 11 управо: $(3584)_{11}$.

•$(B48)_{14} \xrightarrow{?} (x)_{11}$

Развој броја $(B48)_{14}$ је: $B \cdot 14^2 + 4 \cdot 14 + 8$

Да бисмо у овој ситуацији применили директан алгоритам, потребно је све бројеве у претходном изразу пребацити у основу 11 (у претходном примеру за тим није било потребе, јер је основа 7 мања од основе 11).

Како је $(B)_{14} = (10)_{11}$ и $(14)_{10} = (13)_{11}$, посматрамо следећи израз: $10 \cdot 13^2 + 4 \cdot 13 + 8 = (10 \cdot 13 + 4) \cdot 13 + 8$ који израчунавамо у основи 11:

$$(10 \cdot 13 + 4) \cdot 13 + 8 = (130 + 4) \cdot 13 + 8 = 134 \cdot 13 + 8 = 1731 + 8 = 1739$$

Одакле добијамо да је тражени запис броја $(B48)_{14}$ у основи 11: $(1739)_{11}$.

•Збир ова два броја у основи 11 је: $(3584)_{11} + (1739)_{11} = (5212)_{11}$.

2. ЗАДАТАК

•Како је (+7) позитиван број, његов запис у бинарном потпуном комплементу помоћу 4 цифре биће 0111. Тражени запис броја (-7) биће: $10000 - 0111 = 1001$.

•Сада је потребно израчунати $1001 \cdot 0111$. Ово можемо да урадимо на три начина.

1. начин: Помноже се апсолутне вредности бројева чији су дати записи. Резултату, уколико има потребе, променити знак.
Запис апсолутне вредности првог чиниоца јесте 0111, а другог такође 0111 (користимо, дакле, 4 цифре за запис).
Сада је потребно помножити $0111 \cdot 0111$:

$$
\begin{array}{rr}
0111 \cdot 0111 = & 0111 \\
 & 0111 \\
 & 0111 \\
+ & 0000 \\
\hline
 & 0110001
\end{array}
$$

Како је запис првог чиниоца 1001 - негативан запис, а запис другог чиниоца 0111 - ненегативан запис, потребно је овом резултату множења апсолутних вредности чинилаца променити знак (односно крајњи резултат је негативан).
Дакле, резултат записан у потпуном комплементу помоћу 7 цифара јесте:

$$10000000 - 0110001 = 1001111.$$

2. начин (помоћу хардверског алгоритма за неозначене бројеве): Опет посматрамо апсолутне вредности чинилаца и помоћу хардверског алгоритма вршимо њихово множење. Дакле, множимо 0111 и 0111. Множеник (први чинилац) је 0111 и њега уписујемо у регистар M, док множилац (други чинилац) 0111 уписујемо у регистар P. Иницијално, у регистар A уписујемо 0000, а у регистар C 0.

Корак	C	A	P	Коментар
0	0	0000	0111	Иницијални садржаји регистара. Садржај регистра M је: 0111.
1	0	0111	0111	Последња цифра регистра P је била 1 => врши се сабирање A=A+M.
	0	0011	1011	Врши се логичко померање удесно регистра CAP.
2	0	1010	1011	Последња цифра регистра P је била 1 => врши се сабирање A=A+M.
	0	0101	0101	Врши се логичко померање удесно регистра CAP.
3	0	1100	0101	Последња цифра регистра P је била 1 => врши се сабирање A=A+M.
	0	0110	0010	Врши се логичко померање удесно регистра CAP.
4	0	0011	0001	Врши се логичко померање удесно регистра CAP.

Резултат множења апсолутних вредности читамо из регистра AP: 00110001. Крајњи резултат добијамо када овом резултату променимо знак, тако да ће то бити запис 11001111, помоћу 8 цифара.

3. начин (помоћу Бутовог алгоритма): У овом случају не множимо апсолутне вредности бројева чији су нам записи дати, већ можемо да радимо са самим тим бројевима.
Множеник опет уписујемо у регистар M, множилац у регистар P. У регистар P_{-1} у почетку уписујемо 0.

Корак	A	P	P_{-1}	Коментар
0	0000	0111	0	Иницијални садржаји регистара. Садржај регистра M је: 1001.
1	0111	0111	0	P_0P_{-1}= 10 => врши се одузимање A=A-M.
	0011	1011	1	Врши се аритметичко померање удесно регистра APP_{-1}.
2	0011	1011	1	P_0P_{-1}= 11 => не врши се никаква акција.
	0001	1101	1	Врши се аритметичко померање удесно регистра APP_{-1}.
3	0001	1101	1	P_0P_{-1}= 11 => не врши се никаква акција.
	0000	1110	1	Врши се аритметичко померање удесно регистра APP_{-1}.
4	1001	1110	1	P_0P_{-1}= 01 => врши се сабирање A=A+M.
	1100	1111	0	Врши се аритметичко померање удесно регистра APP_{-1}.

Резултат множења читамо из регистра AP: 11001111.

•Када запис у потпуном комплементу 11001111 пребацимо у основу 10 добијемо да је резултат једнак -49.

3. ЗАДАТАК

•Запис броја +45.24 у основи 2 је +101101.0011110101110000101. Односно, $+1.01101001111010111000010 1 \cdot 2^5$. Из нормализованог облика овог броја можемо да добијемо запис у двострукој тачности према стандарду IEEE-754.
- Знак броја је +, стога је бит знака 0.
- Експонент је 5, у запису са увећањем 1023 запис експонента одговара бинарном запису броја 1023+5=1028 помоћу 11 цифара. Дакле, тражени запис експонента је: 10000000100.
- За запис разломљеног дела имамо 52 цифре.

Из овога добијамо тражени запис броја +45.24:

0 10000000100 0110100111101011100001010000000000000000000000000000

а) 0 10000110 10010001100000000000000

Знак за бит је 0, па је знак броја +. Експонент у запису са увећањем 127 је 10000110, односно број 134, па је експонент 134-127 =7.
Дакле, дати запис одговара броју $+1.100100011 \cdot 2^7$, што је број +200.75 у декадном систему.

б) 0 11111111 00100011111110000001110

Како је експонент максималан, односно 11111111, и како је први бит фракције 0, а остали битови нису сви нула, овај запис представља запис сигналног NaN (SNaN).

в) 1 00000000 00110001100000000000000

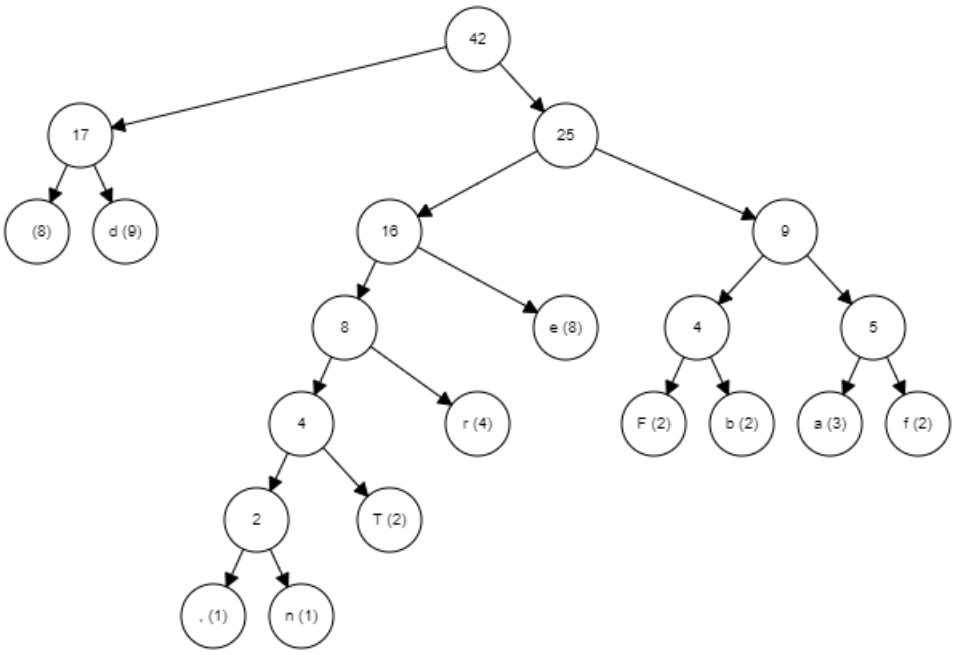
Знак за бит је 1, па је знак броја -. Како су сви битови експонента једнаки 0, дати запис представља запис денормализованог броја. То значи да је декадна вредност експонента -126, а уместо водеће јединице фракције подразумева се водећа нула.
Дакле, ово је запис броја $(-0.001100011)_2 \cdot 2^{-126}$.

4. ЗАДАТАК

Дат је текст "**Fred fed Ted bread, and Ted fed Fred bread**". Сортирана таблица фреквенција карактера је:

Карактер	Фреквенција
'd'	9
'e'	8
' '	8
'r'	4
'a'	3
'f'	2
'T'	2
'b'	2
'F'	2
'n'	1
','	1

Помоћу ове листе формира се једно Хафманово стабло:



Одакле добијамо следеће кодове:

Карактер	Код
’d’	01
’e’	101
’ ’	00
’r’	1001
’a’	1110
’f’	1111
’T’	10001
’b’	1101
’F’	1100
’n’	100001
’,’	100000

Па је код текста "**Fred fed Ted bread, and Ted fed Fred bread**": **1100100110101001111101010010001101010011011001101111001100000001110100001 0100100011010100111110101001100100110101001101100110111 1001** и за његово кодирање су нам потребна 132 бита.

5. ЗАДАТАК

Одговарајућа бинарна форма полинома генератора $G(x) = x^3 + 1$ је 1001.

Дакле,
10001110101
<u>1001</u>
00011110101
  <u>1001</u>
  01100101
   <u>1001</u>
   0101101
    <u>1001</u>
    001001
      <u>1001</u>
      0000

Како је остатак 0, није дошло до грешке приликом преноса дате поруке.

6. ЗАДАТАК

Логичка функција f је задата таблично.

a	b	c	d	$f(a,b,c,d)$
0	0	0	0	1
0	0	0	1	1
0	0	1	0	?
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

Одавде лако формирамо Карноову мапу:

ab \ cd	00	01	11	10
00	1	1	0	?
01	0	0	1	0
11	0	0	1	0
10	1	1	0	1

Непознату вредност на пољу 0010 третирамо као вредност 1, јер нам то омогућава веће заокруживање вредности на пољу 1010.

Прво заокруживање чине поља: 0000, 0010, 1000, 1010.
Друго заокруживање чине поља: 0111, 1111
Треће заокруживање чине поља: 0000, 0001, 1000, 1001

Израз у ДНФ-у који одговара овој мапи ће онда бити:

$$f(a,b,c,d) = (\neg b \wedge \neg d) \vee (b \wedge c \wedge d) \vee (\neg b \wedge \neg c)$$