Лабораторна робота №1

1.1 Тема: Дослідження логічних елементів, логічні та арифметичні операції в двійковій системі

Мета роботи: знайомство з одно- та двомісними функціями, придбання практичних навичок з синтезу комбінаційних схем, засвоєння форм представлення цифрової інформації та операції з її перетворення.

Обладнання: комплект логічних елементів, блок живлення, логічний пробник.

1.2 Теоретичні відомості

Числа, кодування і арифметична інформація.

Двійкові числа. Цифрові обчислювальні машини працюють з двійковими числами. Двійкова система зчислення або система з основою 2 використовує тільки цифри 0 і 1. Ці двійкові числа називаються бітами (від *bi*nary digi*t*). З фізичної точки зору в цифрових електронних системах біт 0 представлений напругою LOW (низьким), а біт 1 – напругою HIGH (високим).

Людська діяльність припускає використання десяткової системи зчислення. Десяткова система, або система із основою 10, містить 10 цифр (від 0 до 9).

Двійковій системі притаманна властивість врівноважування. Двійковому числу 1001_2 (читається: один, нуль, нуль, один) еквівалент 9_{10} в десятковій. Біт одиниці двійкового числа називається молодшим бітом (МБ), біт вісімки – старшим бітом (СБ).

Як перетворити двійкове число 1011 0110 (тобто: один, нуль, один, один, нуль, один, один, нуль) в його десятковий еквівалент? Процедура перетворення виконується у відповідності з табл.1.1. Десяткові значення кожної позиції записані під кожним бітом, потім десяткові числа підсумовуються (128+32+16+4+2=182), що дає 182.

Таблиця 1.1 – Двійково-десяткові перетворення.

Степінь основи	2^7	2^6	2^5		2^4	2^3	2^2		2^1	2^0	
Значення позицій	128	64	32		16	8	4		2	1	
Двійкові	1	0	1		1	0	1		1	0	
Десяткові	128	+	32	+	16	+	4	+	2	=	182

Зазвичай основа системи зчислення вказується індексами. Таким чином, число $1011\ 0110_2$ є двійковим (тому що основа 2), а число 182_{10} – десятковим: $101101101_2 = 182_{10}$.

Як перетворити десяткове 155 в його двійковий еквівалент? Процедура перетворення приведена на рисунку 1.1.

Десяткове спочатку ділиться на 2, що нам дає часткове 77 залишок 1. Цей залишок стає МБ двійкового числа і розміщується в цю позицію (див. рисунок 1.1.). Потім часткове (77) переміщується, як показує стрілка, і стає наступним, що буде ділитися. Потім кожне часткове поступово ділиться на 2 до тих пір, поки не отримаємо часткове, рівне 0, і залишок рівний 1 (див.

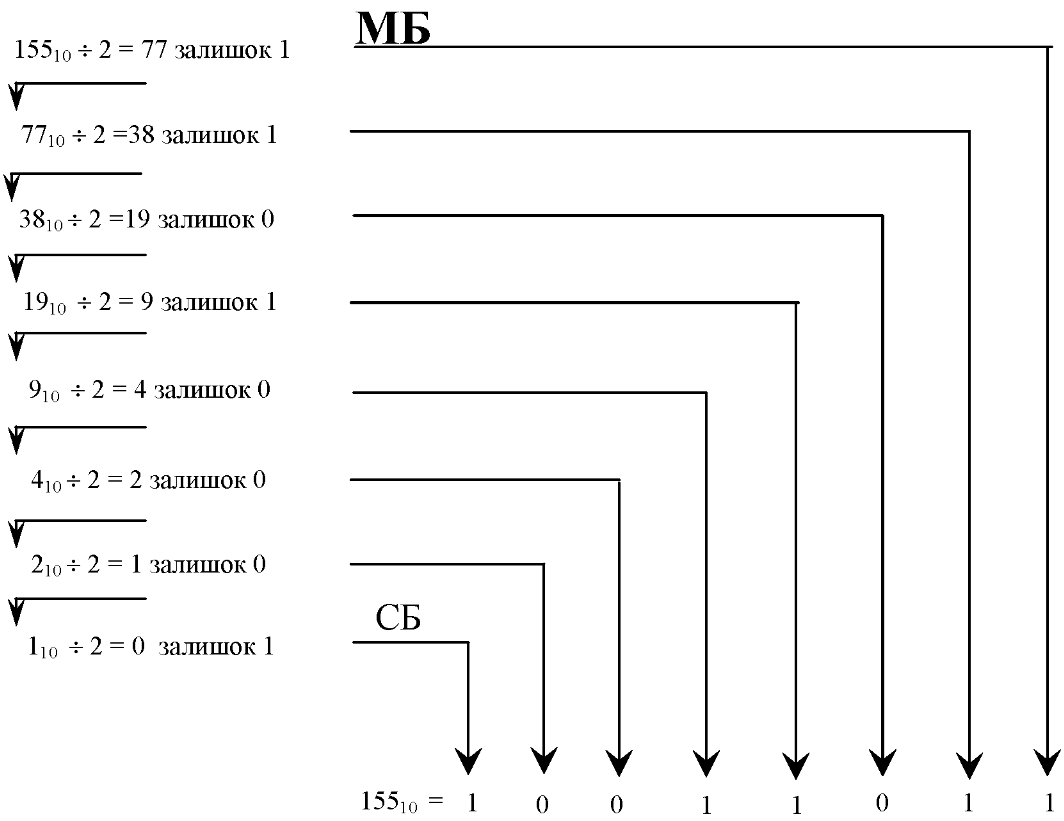


Рисунок 1.1 — Двійково-десяткові перетворення

останній рядок на рис. 1.1). Останній рядок дає нам результат $155_{10}=10011011_2$.

Шістнадцяткові числа.

Комірка пам'яті мікро – ЕОМ може містити двійкове число 10011110. Такий довгий ланцюжок нулів і одиниць складний для запам'ятовування і незручний для введення з клавіатури. Число 1001 1110 могло б бути перетворене в десяткове, що дало б 158_{10}, але процес перетворення зайняв би багато часу. Більша частина систем мікроінформатики використовують шістнадцяткову форму запису, щоб спростити запам'ятовування і використання таких двійкових чисел, як 1001 1110.

Шістнадцяткова система зчислення (*hexadecimal*), або система із основою 16, використовує 16 символів від 0 до 9 і A, B, C, D, E, F. Потрібно зауважити, що кожний шістнадцятковий символ може бути представлений єдиним сполученням чотирьох бітів. Таким чином, представленням двійкового числа 1001 1110 в шістнадцятковому коді є число 9E. Це значить, що частина 1001 двійкового числа дорівнює 9, а друга частина 1110 дорівнює E (звичайно, в шістнадцятковому коді). Звідси, $1001\ 1110_2 = 9E_{16}$. (не слід забувати, що індекси означають основу системи зчислення).

Як перетворити двійкове число 0111010 в шістнадцяткове? Потрібно розпочати з МБ і розділити двійкове число на групи із 4 бітів. Потім потрібно замінити кожну групу із 4 бітів еквівалентною шістнадцятковою цифрою: $1010_2 = A$, $0011_2 = 3$, звідси, $111010_2 = 3\ A_{16}$.

Як перетворити шістнадцяткове число 7F в двійкове? В цьому випадку кожна шістнадцяткова цифра повинна бути замінена своїм двійковим еквівалентом з 4 біт. В прикладі двійкове число 0111 замінене шістнадцятковою цифрою 7, а 1111_2 замінене F_{16}, звідки $7F_{16} = 01111111_2$.

Перетворимо шістнадцяткове число 2C6E в десяткове. Процедурі дій перетворення відповідає табл. 1.2. Значення позицій перших чотирьох шістнадцяткових цифр є, відповідно, зліва направо 4096, 256, 16 і 1. Десяткове число містить 14 (E_{16}) одиниць, 6 чисел 16, 12 (C_{16}) чисел 256 і 2 числа 4096. Кожна цифра множиться на відповідну її вагу, одержуємо суму, яка і дає нам десяткове число 11374.

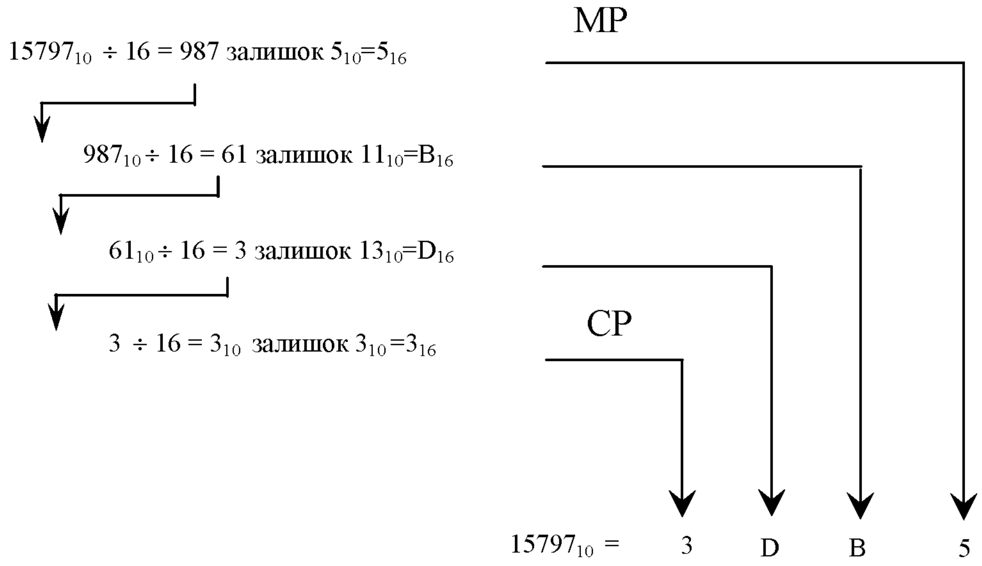
Таблиця 1.2 — Перетворення шістнадцяткового числа в десяткове

Степінь шістнадцяти	16^3		16^2		16^1		16^0	
Значення позиції	4096		256		16		1	
Шістнадцяткове	2		C		6		E	
	4096		256		16		1	
Десяткове	×		×		×		×	
	$\underline{2}$ 8192	+	$\underline{12}$ 3072	+	$\underline{6}$ 96	+	$\underline{14}$ 14	$=1137_{10}$

Перетворимо десяткове число 15797 в шістнадцяткове. На рис. 1.2. показна процедура дій. В першому рядку 15797_{10} розділено на 16, що дає часткове 987_{10} і залишок 5_{10} який потім перетвориться в свій шістнадцяткових еквівалент ($5_{10}=5_{16}$) і стає цифрою молодшого розряду (МР) шістнадцяткового числа. Перше часткове (987) ділиться в другому рядку і знову ділиться на 16, що дає часткове і залишок 11_{10} або шістнадцяткове B. В третьому рядку 61 ділиться на 16, дає часткове 3 і залишок 13_{10} або D_{16} , а в четвертому рядку ділиться 3 на 16, дає часткове 0 і залишок 3_{10} або 3_{16}. Коли часткове рівне 0, як в четвертому рядку, перетворення закінчується. 3_{16} стає цифрою старшого розряду (СР) результату, тобто, $3DB5_{16}$.

Двійково-десяткові числа

З метою зручності перетворення чисті двійкові числа представляються десятковими або шістнадцятковими. Однак, двійково-десяткове перетворення – операція не проста. В калькуляторах, магістралях і числових приладах, коли на доступних користувачу виходах і входах широко розповсюджені десяткові числа, для їх представлення використовують спеціальний



ʼРисунок 1.2 — Десятково-шістнадцяткове перетворення

двійково-десятковий код (ДДК). В табл. 1.3 наведено декілька десяткових чисел і відповідних їм двійково-десяткових еквівалентів (система 8421). Цим визначаються ваги позицій кожного з чотирьох бітів ДДК (використовують інші ДДК, наприклад 5421 і плюс 3).

Таблиця 1.3 – Двійково-десятковий код

Десяткове число	Двійково-десяткові числа			
	8	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

Запишемо десяткове число 3691 в ДДК 8421. Кожна десяткова цифра перетворюється прямо в свій двійково-десятковий еквівалент із 4 бітів, і перетворення дають 3691_{10}=$0011\ 0110\ 1001\ 0001_{ДДК}$.

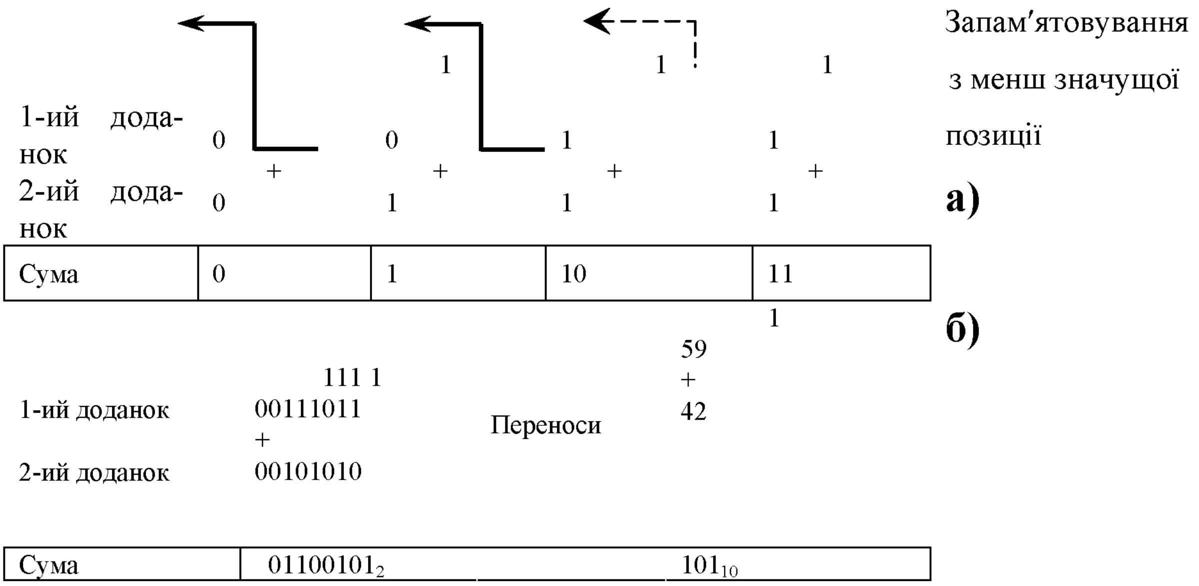
Перетворимо тепер двійково-десяткове число 1000000001110010 в його десятковий еквівалент. Кожна група із 4 бітів прямо перетворюється в її десятковий еквівалент, і тоді отримуємо $1000\ 0000\ 0111\ 0010_{ДДК} = 8072_{10}$:

Мікропроцесори складають чисті двійкові числа, але вони мають, однак, команди для перетворення результату своїх складань в двійково-десяткове записування. Отримані двійково-десяткові числа легко потім представити в десятковому записі, використовуючи прості процедури, що були описані вище.

Двійкова арифметика

Додавання, віднімання або множення двійкових чисел виконується так само, як і в арифметиці двійкових чисел. Більшість мікропроцесорів мають команди додавання і віднімання двійкових чисел, однак деякі, менш багаточисельні виконують команди множення і ділення (наприклад, мікропроцесори INTEL 8086 і INTEL 8088).

На рис.1.3, а запропоновані прості правила двійкового додавання. Два перших (зліва) правила очевидні, третє показує, що 1+1=10, тобто, найбільш значуща 1 переноситься в ближчий старший розряд. Четверте правило, на кінець, показує, що 1+1+1=11. В цьому випадку перший, другий доданки і запам’ятовувальне в результаті додавання в молодшому розряді число – все 1.



а – правило; б – приклад

Рисунок 1.3 – Двійкове додавання

Додамо двійкові числа 0011 1011 і 0010 1010 (операція показана на рис.1.4,б). Для великої ясності дії з десятковими еквівалентами, числа що оброблюються, показані на рисунку справа. Сумою двох чисел 0011 1011і 0010 1010 буде 0110 0101_2.

На рис.1.4, а наведені правила двійкового віднімання. Перші три аналогічні десятковому розрахуванню. Останнє потребує займу з більш значущого попереднього розряду (в цьому випадку вага 2). Зменшувальним є двійкове число 10, від′ємником 1, різницею – 1.

					1 10	
				10	010 010	
Зменшува-льне	_0	_1	_1	_10	_01010101	_85
Від'ємник	0	0	1	1	00111001	57
Різниця	1	1	0	1	00011100_2	28_{10}

а) **б)**

а) – правила; б) – приклад

Рисунок 1.4 – Двійкове віднімання

Віднімаємо двійкове число 0011 1001 від 0101 0101. Цей приклад наведений на рис.1.4, б. Розряди ваг 1, 2 і 4 цього двійкового обчислення прості для виконання і відносяться до перших трьох правил на рис.1 .5, а. В колонці ваги 8 має місце віднімання 1 з 0. Тоді 1 запозичується із колонки ваги 16. Одиниця віднімається із 10_2, що дає різницю 1 згідно з четвертим правилом на рис. 1.5, а. Після цього запозичення в колонці ваги 16 має місце віднімання 1 з 0. Згідно з четвертим правилом 1 повинна бути запозичена із наступної, більш значущої позиції (колонка ваги 32), але в колонці 32 маємо 0; тому колонка 32 повинна запозичити з колонки ваги 64, що і зроблено. На кінець колонка 16 стає 10_2, від'ємник 1, різниця 1. В колонці 32 маємо 1-1=0, в колонці 64 – 0-0=0, в колонці 128 – 0-0=0. таким часом, рис. 1.5,б ілюструє операцію обчислення $0011\ 1001_2$ із $0101\ 0101_2$ (справа ця задача вирішена в десятковому записі).

Наведемо правила десяткового множення:

Множене	0	1	0	1
	×	×	×	×
Множники	0	0	1	1
Добутки	0	0	0	1

Два перших правила не потребують пояснення. В двох наступних множником є 1: коли множником є 1 при двійковому множенні, множене стає результатом і представляє собою добуток. Коли множник 0, добуток завжди 0.

Виконаємо множення 1101 на 101. Як і у випадку множення десяткових чисел, множене спочатку множиться на число, що стоїть в молодшому розряді (у випадку, що розглядається – біт в колонці ваги 1).

Оскільки біт множника в розряді ваги 1 є 1, множене копіюється і складає перший частковий добуток. Другим бітом множника є 0, тоді другий частковий добуток дорівнює 0000 (він зсунутий на одну позицію вліво). Бітом розряду ваги 4-го множника є 1, тоді для отримання третього часткового добутку знову слідує копіювання множеного (копіювання завершується новим зсувом на одну позицію вліво). Після цього виконуємо додавання трьох часткових добутків, що дає результат $1101_2 \times 101_2 = 1000001_2$ відповідає добутку десяткових чисел $13_{10} \times 5_{10} = 65_{10}$.

Множене	1101	13
	×	×
Множник	101	5
1-ий частковий добуток	1101	65_{10}
2-ий частковий добуток	0000	
3-ій частковий добуток	1101	
Кінцевий добуток	1000001_2	

Додатковий код.

Сама ЕОМ оброблює інформацію, зазвичай, в двійковому коді. Однак, якщо потрібно використовувати цифри із знаком, використовують спеціальний додатковий код, що спрощує апаратні засоби ЕОМ.

Звичайний регістр МП представляють простором із 8 бітів даних. Позиції бітів пронумеровані від 7 до 0, а ваги двійкових позицій вказані в основі регістра, біт 7 має вагу 128, біт 8 – 64 і так далі.

В обох випадках біт 7 є знаковим. Він показує, чи є число додатним (+) або від'ємним (–). При 0 в знаковому біті число додатне, при 1 – від'ємне.

Якщо число додатне, ті комірки пам'яті (6-0), що зостались, містять двійкове 7–розрядне число. Наприклад, якщо регістр містить 0100 0001, це відповідає числу $+65_{10}$ (64+1, знаковий біт додатний). Якщо в нього записано 0111 1111, буде містити $+127_{10}$ (знаковий біт додатний: 0+64+32+16+8+4+2+1), що є найбільшим додатним числом, яке може містити 7-розрядний регістр.

В табл. 1.4 наведений запис в додатковому коді додатних та від'ємних чисел. Всі додатні числа мають 0 в старшому біті, інші біти складають двійкове число. Всі від'ємні числа мають 1 в старшому розряді. Розглянемо рядок +0 в табл. 1.4: запис в додатковому коді +0 буде 0000 0000. В найближчому нижньому рядку бачимо, що запис в додатковому коді – 1 такий: 1111 1111. Розглянемо покрокове переміщення в зворотному напрямку від 0000 0000 до 1111 1111.

Таблиця 1.4 — Десяткові числа із знаками і їх представлення в додатковому коді

Десяткові	Представлення чисел із знаками	Примітки
+127	0111 1111	
•	•	
•	•	
•	•	
+8	0000 1111	
+7	0000 0111	Додатні числа представлені в тій же формі, що і прямі двійкові числа.
+6	0000 0110	
+5	0000 0101	
+6	0000 0100	
+3	0000 0011	
+2	0000 0010	
+1	0000 0001	
+0	0000 0000	
-1	1111 1111	
-2	1111 1110	
-3	1111 1101	
-4	1111 1100	
-5	1111 1011	
-6	1111 1010	Від'ємні числа представлені у формі додаткового коду.
-7	1111 1001	
-8	1111 1000	
•	•	
•	•	
•	•	
-128	1000 0000	

Який буде запис в додатковому коді числа –9? Розглянемо етапи перетворення. Вони наведені в таблиці 1.5.

Одержаний результат є додатковим кодом додатного десяткового числа. В наведеному прикладі додатковим кодом числа 9 є 1111 0111. Потрібно замітити, що знаковий біт 1, це означає, що дане число (1111 0111) від'ємне.

Яким буде десятковий еквівалент числа 1111 0000, що записаний у формі додаткового коду? Процедура в цьому випадку наведена в таблиці 1.6.

Таблиця 1.5 — Запис в додатковому коді числа мінус 9

Десяткове число	9	Етап 1.	Запис десяткового числа без знаку (9).
Двійкове число	0000 1001	Етап 2.	Перетворення десяткового числа в двійковий код (0000 1001).
Доповнення до 1 (зворотний або інверсний код)	1111 0110	Етап 3.	Отримання зворотного коду двійкового числа заміною нулів одиницею, а одиниць – нулями (1111 0110).
Доповнення до 2 (додатковий код)	+1 1111 0111	Етап 4.	Додати одиницю до зворотного коду. Тут додати 1 до 1111 0110, що дає 1111 0111.

Таблиця 1.6 — Десятковий еквівалент числа 1111 0000

Додатковий код	1111 0000	Етап 1.	Запис додаткового коду (1111 0000).
Доповнення до 1	0000 1111	Етап 2.	Утворюється зворотний код додаткового коду заміною нулів одиницями, а одиниць – нулями (0000 1111).
Двійкове число	+1 0001 0000 = 16	Етап 3.	Додати 1.

Таким чином, формування зворотного коду і додавання 1 є тими ж процедурами, які ми проводили при перетворенні двійкового числа в додатковий код. Однак, слід відзначити, що хоча ми отримали двійкове число 0001 0000 = 16_{10}, вихідний запис додаткового коду 1111 0000 = -16, тобто, маємо від’ємне число, оскільки старший біт в додатковому коді є 1.

1.3 Послідовність виконання роботи та зміст звіту

Результати роботи з логічних функцій оформлюються у вигляді таблиць істинності (відповідності);

Результати з перетворення цифрової інформації оформлюються у вигляді таблиці результатів обчислень:

Таблиця 1.7 — Результати обчислень

Вихідне число	Z=	Двійкова форма									Десятковий еквівалент
1	2	3	4	5	6	7	8	9	10	11	12
Двійковий код	Z_2										

Продовження таблиці 1.7

1	2	3	4	5	6	7	8	9	10	11	12
Зсув на розряд вліво	Z_3										
Обернений код	Z_4										
Доповняльний код	Z_5										
Арифметична сума	Z_2+Z_3										
Арифметична сума	Z_2+Z_2										
Арифметична сума	Z_2+Z_5										
Логічна сума	Z_2+Z_2										
Логічна сума	Z_2+Z_3										
Логічна сума	Z_2+Z_5										
Порозрядна сума за мод. два	$Z_2 \oplus Z_2$										
Порозрядна сума за мод. два	$Z_2 \oplus Z_3$										
Логічне множення	$Z_2 * Z_2$										
Логічне множення	$Z_2 * Z_3$										
Логічне множення	$Z_2 * Z_5$										

1.4 Варіанти завдань

№ вар.	X_{10}	№ вар.	X_{10}
1	101	14	86
2	98	15	115
3	103	16	84
4	96	17	117
5	105	18	82
6	94	19	119
7	107	20	80
8	92	21	121
9	109	22	78
10	90	23	123
11	111	24	76
12	88	25	125
13	113		