

# Skaičiavimo sistemos. Dvejetainė aritmetika

Saulius Gražulis

Vilnius, 2020

Vilniaus universitetas, Matematikos ir informatikos fakultetas  
Informatikos institutas



Ši skaidrių rinkinį galima kopijuoti, kaip nurodyta Creative Commons  
[Attribution-ShareAlike 4.0 International](https://creativecommons.org/licenses/by-sa/4.0/) licenzijoje



# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0			

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0			
	1			

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0			
	1			
	2			

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0			
	1			
	2			
	3			

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0			
	1			
	2			
	3			
	4			
	5			
	6			
	7			
	8			
	9			

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0			
	1			
	2			
	3			
	4			
	5			
	6			
	7			
	8			
	9			
	<b>10</b>			



# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0			
	1			
	2			
	3			
	4			
	5			
	6			
	7			
	8			
	9			
	10			
	11			

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0			
	1			
	2			
	3			
	4			
	5			
	6			
	7			
	8			
	9			
	10			
	11			
	12			
	13			
	14			
	15			
	16			
	17			

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0		
	1			
	2			
	3			
	4			
	5			
	6			
	7			
	8			
	9			
	10			
	11			
	12			
	13			
	14			
	15			
	16			
	17			

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0		
	1	1		
	2			
	3			
	4			
	5			
	6			
	7			
	8			
	9			
	10			
	11			
	12			
	13			
	14			
	15			
	16			
	17			

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0		
	1	1		
	2	<b>10</b>		
	3			
	4			
	5			
	6			
	7			
	8			
	9			
	10			
	11			
	12			
	13			
	14			
	15			
	16			
	17			

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0		
	1	1		
	2	10		
	3	11		
	4			
	5			
	6			
	7			
	8			
	9			
	10			
	11			
	12			
	13			
	14			
	15			
	16			
	17			

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparatūroje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0		
	1	1		
	2	10		
	3	11		
	4	<b>100</b>		
	5			
	6			
	7			
	8			
	9			
	10			
	11			
	12			
	13			
	14			
	15			
	16			
	17			

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0		
	1	1		
	2	10		
	3	11		
	4	100		
	5	101		
	6			
	7			
	8			
	9			
	10			
	11			
	12			
	13			
	14			
	15			
	16			
	17			



# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0		
	1	1		
	2	10		
	3	11		
	4	100		
	5	101		
	6	110		
	7			
	8			
	9			
	10			
	11			
	12			
	13			
	14			
	15			
	16			
	17			

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0		
	1	1		
	2	10		
	3	11		
	4	100		
	5	101		
	6	110		
	7	111		
	8			
	9			
	10			
	11			
	12			
	13			
	14			
	15			
	16			
	17			

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0		
	1	1		
	2	10		
	3	11		
	4	100		
	5	101		
	6	110		
	7	111		
	8	<b>1000</b>		
	9			
	10			
	11			
	12			
	13			
	14			
	15			
	16			
	17			

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0		
	1	1		
	2	10		
	3	11		
	4	100		
	5	101		
	6	110		
	7	111		
	8	1000		
	9	1001		
	10			
	11			
	12			
	13			
	14			
	15			
	16			
	17			

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0		
	1	1		
	2	10		
	3	11		
	4	100		
	5	101		
	6	110		
	7	111		
	8	1000		
	9	1001		
	10	1010		
	11	1011		
	12	1100		
	13	1101		
	14	1110		
	15	1111		
	16	10000		
	17	10001		

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0	0	
	1	1	1	
	2	10	2	
	3	11	3	
	4	100	4	
	5	101	5	
	6	110	6	
	7	111	7	
	8	1000		
	9	1001		
	10	1010		
	11	1011		
	12	1100		
	13	1101		
	14	1110		
	15	1111		
	16	10000		
	17	10001		

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0	0	
	1	1	1	
	2	10	2	
	3	11	3	
	4	100	4	
	5	101	5	
	6	110	6	
	7	111	7	
	8	1000	10	
	9	1001		
	10	1010		
	11	1011		
	12	1100		
	13	1101		
	14	1110		
	15	1111		
	16	10000		
	17	10001		

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparatūroje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0	0	
	1	1	1	
	2	10	2	
	3	11	3	
	4	100	4	
	5	101	5	
	6	110	6	
	7	111	7	
	8	1000	10	
	9	1001	11	
	10	1010	12	
	11	1011	13	
	12	1100	14	
	13	1101	15	
	14	1110	16	
	15	1111	17	
	16	10000	20	
	17	10001	21	



# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparatūroje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0	0	0
	1	1	1	1
	2	10	2	2
	3	11	3	3
	4	100	4	4
	5	101	5	5
	6	110	6	6
	7	111	7	7
	8	1000	10	8
	9	1001	11	9
	10	1010	12	
	11	1011	13	
	12	1100	14	
	13	1101	15	
	14	1110	16	
	15	1111	17	
	16	10000	20	
	17	10001	21	

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0	0	0
	1	1	1	1
	2	10	2	2
	3	11	3	3
	4	100	4	4
	5	101	5	5
	6	110	6	6
	7	111	7	7
	8	1000	10	8
	9	1001	11	9
	10	1010	12	A
	11	1011	13	
	12	1100	14	
	13	1101	15	
	14	1110	16	
	15	1111	17	
	16	10000	20	
	17	10001	21	

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparatūroje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0	0	0
	1	1	1	1
	2	10	2	2
	3	11	3	3
	4	100	4	4
	5	101	5	5
	6	110	6	6
	7	111	7	7
	8	1000	10	8
	9	1001	11	9
	10	1010	12	A
	11	1011	13	B
	12	1100	14	
	13	1101	15	
	14	1110	16	
	15	1111	17	
	16	10000	20	
	17	10001	21	

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0	0	0
	1	1	1	1
	2	10	2	2
	3	11	3	3
	4	100	4	4
	5	101	5	5
	6	110	6	6
	7	111	7	7
	8	1000	10	8
	9	1001	11	9
	10	1010	12	A
	11	1011	13	B
	12	1100	14	C
	13	1101	15	
	14	1110	16	
	15	1111	17	
	16	10000	20	
	17	10001	21	

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparatūroje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0	0	0
	1	1	1	1
	2	10	2	2
	3	11	3	3
	4	100	4	4
	5	101	5	5
	6	110	6	6
	7	111	7	7
	8	1000	10	8
	9	1001	11	9
	10	1010	12	A
	11	1011	13	B
	12	1100	14	C
	13	1101	15	D
	14	1110	16	
	15	1111	17	
	16	10000	20	
	17	10001	21	

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparatūroje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0	0	0
	1	1	1	1
	2	10	2	2
	3	11	3	3
	4	100	4	4
	5	101	5	5
	6	110	6	6
	7	111	7	7
	8	1000	10	8
	9	1001	11	9
	10	1010	12	A
	11	1011	13	B
	12	1100	14	C
	13	1101	15	D
	14	1110	16	E
	15	1111	17	
	16	10000	20	
	17	10001	21	

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0	0	0
	1	1	1	1
	2	10	2	2
	3	11	3	3
	4	100	4	4
	5	101	5	5
	6	110	6	6
	7	111	7	7
	8	1000	10	8
	9	1001	11	9
	10	1010	12	A
	11	1011	13	B
	12	1100	14	C
	13	1101	15	D
	14	1110	16	E
	15	1111	17	F
	16	10000	20	
	17	10001	21	

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0	0	0
	1	1	1	1
	2	10	2	2
	3	11	3	3
	4	100	4	4
	5	101	5	5
	6	110	6	6
	7	111	7	7
	8	1000	10	8
	9	1001	11	9
	10	1010	12	A
	11	1011	13	B
	12	1100	14	C
	13	1101	15	D
	14	1110	16	E
	15	1111	17	F
	16	10000	20	10
	17	10001	21	



# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0	0	0
	1	1	1	1
	2	10	2	2
	3	11	3	3
	4	100	4	4
	5	101	5	5
	6	110	6	6
	7	111	7	7
	8	1000	10	8
	9	1001	11	9
	10	1010	12	A
	11	1011	13	B
	12	1100	14	C
	13	1101	15	D
	14	1110	16	E
	15	1111	17	F
	16	10000	20	10
	17	10001	21	11

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0	0	0
	1	1	1	1
	<b>2</b>	<b>10</b>	2	2
	3	11	3	3
	<b>4</b>	<b>100</b>	4	4
	5	101	5	5
	6	110	6	6
	7	111	7	7
	<b>8</b>	<b>1000</b>	<b>10</b>	8
	9	1001	11	9
	<b>10</b>	1010	12	A
	11	1011	13	B
	12	1100	14	C
	13	1101	15	D
	14	1110	16	E
	15	1111	17	F
	<b>16</b>	<b>10000</b>	20	<b>10</b>
	17	10001	21	11

# Dvejetainė aritmetika

Skaitmenys, kuriuos galime atvaizduoti savo aparaturoje: **0** ir **1**

Sistema: Pagr.:	dec 10	bin 2	oct 8	hex 16
	0	0	0	0
	1	1	1	1
	2	10	2	2
	3	11	3	3
	4	100	4	4
	5	101	5	5
	6	110	6	6
	7	111	7	7
	8	<b>1000</b>	<b>10</b>	8
	9	1001	11	9
	10	1010	12	A
	11	1011	13	B
	12	1100	14	C
	13	1101	15	D
	14	1110	16	E
	15	1111	17	F
	16	<b>10000</b>	<b>20</b>	<b>10</b>
	17	10001	21	11

Sistemoje pagrindu  $b$ ,  $m$  skaitmenų skaičius atvaizduojamas kaip daugianaris:

$$N = a_{m-1}b^{m-1} + a_{m-2}b^{m-2} + \dots + a_2b^2 + a_1b + a_0$$

Pavyzdys:

$$\begin{aligned} 110101_2 &= 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 \\ &= 32_{10} + 16_{10} + 4_{10} + 1_{10} \\ &= 53_{10} \end{aligned}$$

Kitas pavyzdys:

$$16_{16} = 1 \cdot 16^1 + 6 \cdot 16^0 = 16_{10} + 6_{10} = 22_{10}$$

# Trupmenų reiškimas

Sistemoje pagrindu  $b$ , trupmena  $f = 0.a_1 a_2 \dots a_m$  atvaizduojama kaip daugianaris:

$$f = a_1 b^{-1} + a_2 b^{-2} + \dots + a_{m-1} b^{-m+1} + a_m b^{-m}$$

Example:

$$\begin{aligned} 0.0110101_2 &= 0 \cdot 2^{-1} + 1 \cdot 2^{-2} + 1 \cdot 2^{-3} + 0 \cdot 2^{-4} + \\ &\quad + 1 \cdot 2^{-5} + 0 \cdot 2^{-6} + 1 \cdot 2^{-7} \\ &= \frac{1}{4_{10}} + \frac{1}{8_{10}} + \frac{1}{32_{10}} + \frac{1}{128_{10}} \\ &= 0.4140625_{10} \end{aligned}$$

# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$\begin{aligned} N &= a_{m-1}b^{m-1} + a_{m-2}b^{m-2} + \dots + a_2b^2 + a_1b + a_0 \\ &= b(a_{m-1}b^{m-2} + a_{m-2}b^{m-3} + \dots + a_2b + a_1) + a_0 \end{aligned}$$

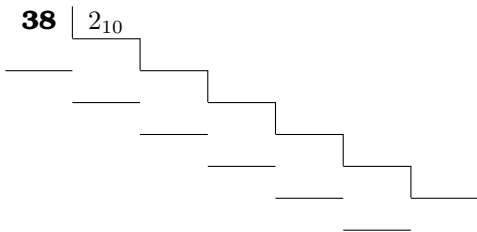
# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:



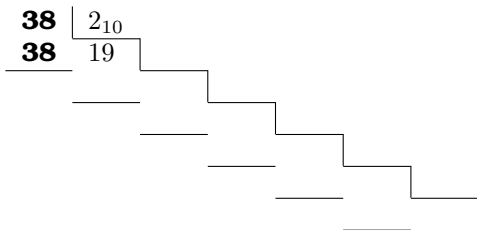
# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:





# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:

$$\begin{array}{r} \mathbf{38} \quad | \quad 2_{10} \\ \mathbf{38} \quad | \quad 19 \\ \hline \mathbf{0} \end{array}$$

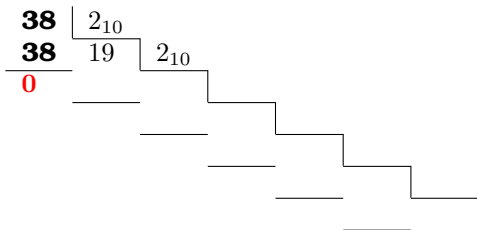
# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:



# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:

<b>38</b>	2 <sub>10</sub>	
<b>38</b>	19	2 <sub>10</sub>
<b>0</b>	18	9

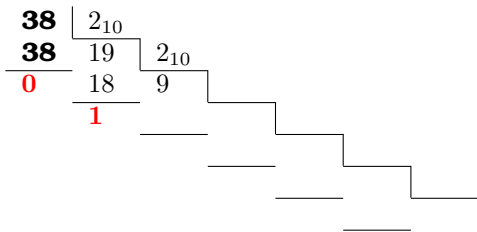
# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:



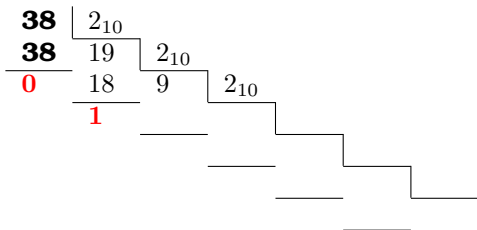
# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:



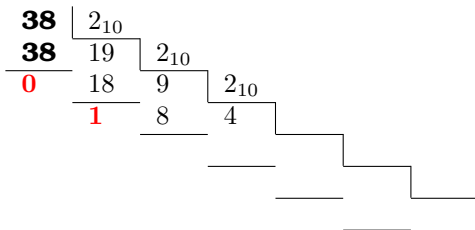
# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:



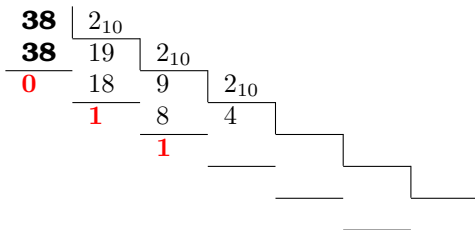
# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:



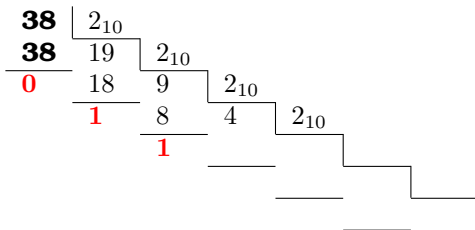
# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:





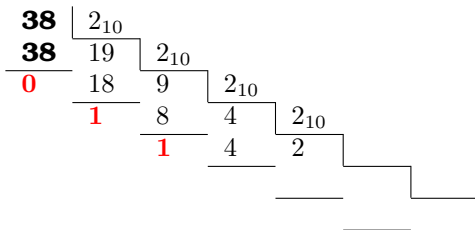
# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:



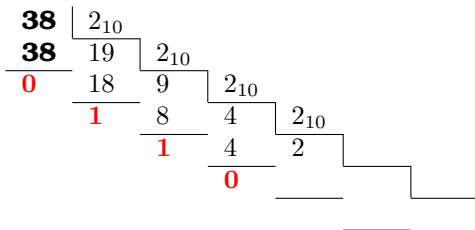
# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:



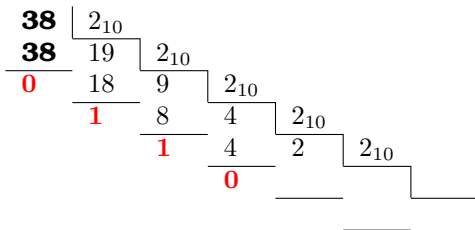
# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:



# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:

<b>38</b>	$2_{10}$								
<b>38</b>	19	$2_{10}$							
<b>0</b>	18	9	$2_{10}$						
	<b>1</b>	8	4	$2_{10}$					
		<b>1</b>	4	2	$2_{10}$				
			<b>0</b>	2	1	$2_{10}$			

# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:

<b>38</b>	$2_{10}$							
<b>38</b>	19	$2_{10}$						
<b>0</b>	18	9	$2_{10}$					
	<b>1</b>	8	4	$2_{10}$				
		<b>1</b>	4	2	$2_{10}$			
			<b>0</b>	2	1	$2_{10}$		
				<b>0</b>	1			

# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:

<b>38</b>	$2_{10}$								
<b>38</b>	19	$2_{10}$							
<b>0</b>	18	9	$2_{10}$						
	<b>1</b>	8	4	$2_{10}$					
		<b>1</b>	4	2	$2_{10}$				
			<b>0</b>	2	1	$2_{10}$			
				<b>0</b>	1	$2_{10}$			
					0	1	$2_{10}$		
						0	1	$2_{10}$	
							0	1	$2_{10}$
								0	1
									0

# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:

<b>38</b>	$2_{10}$								
<b>38</b>	19	$2_{10}$							
<b>0</b>	18	9	$2_{10}$						
	<b>1</b>	8	4	$2_{10}$					
		<b>1</b>	4	2	$2_{10}$				
			<b>0</b>	2	1	$2_{10}$			
				<b>0</b>	0	0	$2_{10}$		
					<b>0</b>	0	0	$2_{10}$	

# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:

<b>38</b>	$2_{10}$								
<b>38</b>	19	$2_{10}$							
<b>0</b>	18	9	$2_{10}$						
	<b>1</b>	8	4	$2_{10}$					
		<b>1</b>	4	2	$2_{10}$				
			<b>0</b>	2	1	$2_{10}$			
				<b>0</b>	0	0	$2_{10}$		
					<b>1</b>				



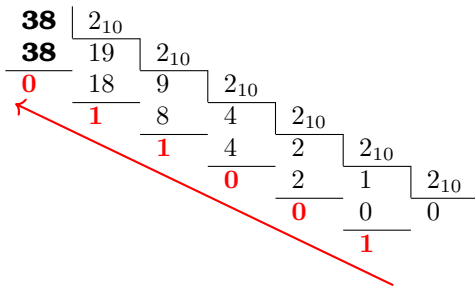
# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:



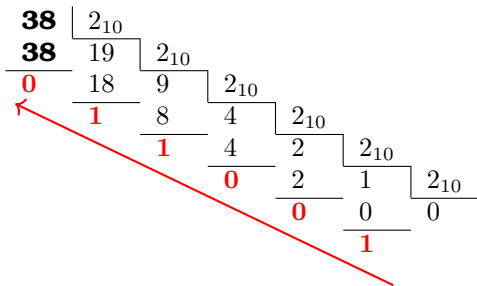
# Pavertimas į kitas skaičiavimo sistemas

Dalyba iš sistemos bazės duoda liekaną, kuri yra jauniausias skaičiaus skaitmuo:

$$100110_2 / 10_2 = 10011.0_2 \quad (\text{quotient} = 10011_2, \text{remainder} = 0)$$

$$10011_2 / 10_2 = 1001.1_2 \quad (\text{quotient} = 1001_2, \text{remainder} = 1)$$

Taigi, pakartotinė dalyba iš sistemos *pagrindo* duoda mums visus skaitmenis toje sistemoje kaip dalybos *liekanas*:



$$100110_2 = 2^5 + 2^2 + 2^1 = 32_{10} + 4_{10} + 2_{10} = \mathbf{38}_{10}$$

# Trupmenų pavertimas

Trupmenas galime padauginti iš skaičiavimo sistemos bazės ir gauti vieną po kito skaitmenis už trupmenos kablelio:

# Trupmenų pavertimas

Trupmenas galime padauginti iš skaičiavimo sistemos bazės ir gauti vieną po kito skaitmenis už trupmenos kablelio:

$$0.01101_2 \times 10_2 = 0.1101_2; \quad \lfloor 0.01101_2 \times 10_2 \rfloor = \lfloor 0.1101_2 \rfloor = \mathbf{0}$$

# Trupmenų pavertimas

Trupmenas galime padauginti iš skaičiavimo sistemos bazės ir gauti vieną po kito skaitmenis už trupmenos kablelio:

$$0.01101_2 \times 10_2 = 0.1101_2; \quad \lfloor 0.01101_2 \times 10_2 \rfloor = \lfloor 0.1101_2 \rfloor = \mathbf{0}$$

$$\mathbf{0}.1101_2 \times 10_2 = 1.101_2; \quad \lfloor 0.1101_2 \times 10_2 \rfloor = \lfloor 1.101_2 \rfloor = \mathbf{1}$$

# Trupmenų pavertimas

Trupmenas galime padauginti iš skaičiavimo sistemos bazės ir gauti vieną po kito skaitmenis už trupmenos kablelio:

$$0.01101_2 \times 10_2 = 0.1101_2; \quad \lfloor 0.01101_2 \times 10_2 \rfloor = \lfloor 0.1101_2 \rfloor = \mathbf{0}$$

$$\mathbf{0}.1101_2 \times 10_2 = 1.101_2; \quad \lfloor 0.1101_2 \times 10_2 \rfloor = \lfloor 1.101_2 \rfloor = \mathbf{1}$$

$$1.101_2 - \lfloor 1.101_2 \rfloor = \mathbf{0}.101_2$$

# Trupmenų pavertimas

Trupmenas galime padauginti iš skaičiavimo sistemos bazės ir gauti vieną po kito skaitmenis už trupmenos kablelio:

$$0.01101_2 \times 10_2 = 0.1101_2; \quad \lfloor 0.01101_2 \times 10_2 \rfloor = \lfloor 0.1101_2 \rfloor = \mathbf{0}$$

$$\mathbf{0}.1101_2 \times 10_2 = 1.101_2; \quad \lfloor 0.1101_2 \times 10_2 \rfloor = \lfloor 1.101_2 \rfloor = \mathbf{1}$$

$$1.101_2 - \lfloor 1.101_2 \rfloor = \mathbf{0}.101_2$$

$$\mathbf{0}.101_2 \times 10_2 = 1.01_2; \quad \lfloor 0.101_2 \rfloor = \mathbf{1}$$

# Trupmenų pavertimas

Trupmenas galime padauginti iš skaičiavimo sistemos bazės ir gauti vieną po kito skaitmenis už trupmenos kablelio:

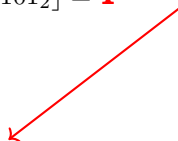
$$0.01101_2 \times 10_2 = 0.1101_2; \quad \lfloor 0.01101_2 \times 10_2 \rfloor = \lfloor 0.1101_2 \rfloor = \mathbf{0}$$

$$\mathbf{0}.1101_2 \times 10_2 = 1.101_2; \quad \lfloor 0.1101_2 \times 10_2 \rfloor = \lfloor 1.101_2 \rfloor = \mathbf{1}$$

$$1.101_2 - \lfloor 1.101_2 \rfloor = \mathbf{0}.101_2$$

$$\mathbf{0}.101_2 \times 10_2 = 1.01_2; \quad \lfloor 0.101_2 \rfloor = \mathbf{1}$$

...





# Trupmenų pavertimas

Trupmenas galime padauginti iš skaičiavimo sistemos bazės ir gauti vieną po kito skaitmenis už trupmenos kablelio:

Paverskime  $0.625_{10}$  į dvejetainę sistemą.

# Trupmenų pavertimas

Trupmenas galime padauginti iš skaičiavimo sistemos bazės ir gauti vieną po kito skaitmenis už trupmenos kablelio:

Paverskime  $0.625_{10}$  į dvejetainę sistemą.

$$0.625_{10} \times 2_{10} = 1.25_{10}; \quad \lfloor 0.625_{10} \times 2_{10} \rfloor = \lfloor 1.25_{10} \rfloor = \mathbf{1}$$

# Trupmenų pavertimas

Trupmenas galime padauginti iš skaičiavimo sistemos bazės ir gauti vieną po kito skaitmenis už trupmenos kablelio:

Paverskime  $0.625_{10}$  į dvejetainę sistemą.

$$0.625_{10} \times 2_{10} = 1.25_{10}; \quad \lfloor 0.625_{10} \times 2_{10} \rfloor = \lfloor 1.25_{10} \rfloor = \mathbf{1}$$

$$1.25_{10} - \lfloor 1.25_{10} \rfloor = 0.25_{10}$$

# Trupmenų pavertimas

Trupmenas galime padauginti iš skaičiavimo sistemos bazės ir gauti vieną po kito skaitmenis už trupmenos kablelio:

Paverskime  $0.625_{10}$  į dvejetainę sistemą.

$$0.625_{10} \times 2_{10} = 1.25_{10}; \quad \lfloor 0.625_{10} \times 2_{10} \rfloor = \lfloor 1.25_{10} \rfloor = \mathbf{1}$$

$$1.25_{10} - \lfloor 1.25_{10} \rfloor = 0.25_{10}$$

$$0.25_{10} \times 2_{10} = 0.5_{10}; \quad \lfloor 0.25_{10} \times 2_{10} \rfloor = \lfloor 0.5_{10} \rfloor = \mathbf{0}$$

# Trupmenų pavertimas

Trupmenas galime padauginti iš skaičiavimo sistemos bazės ir gauti vieną po kito skaitmenis už trupmenos kablelio:

Paverskime  $0.625_{10}$  į dvejetainę sistemą.

$$0.625_{10} \times 2_{10} = 1.25_{10}; \quad \lfloor 0.625_{10} \times 2_{10} \rfloor = \lfloor 1.25_{10} \rfloor = \mathbf{1}$$

$$1.25_{10} - \lfloor 1.25_{10} \rfloor = 0.25_{10}$$

$$0.25_{10} \times 2_{10} = 0.5_{10}; \quad \lfloor 0.25_{10} \times 2_{10} \rfloor = \lfloor 0.5_{10} \rfloor = \mathbf{0}$$

$$0.5_{10} - \lfloor 0.5_{10} \rfloor = 0.5_{10}$$

# Trupmenų pavertimas

Trupmenas galime padauginti iš skaičiavimo sistemos bazės ir gauti vieną po kito skaitmenis už trupmenos kablelio:

Paverskime  $0.625_{10}$  į dvejetainę sistemą.

$$0.625_{10} \times 2_{10} = 1.25_{10}; \quad \lfloor 0.625_{10} \times 2_{10} \rfloor = \lfloor 1.25_{10} \rfloor = \mathbf{1}$$

$$1.25_{10} - \lfloor 1.25_{10} \rfloor = 0.25_{10}$$

$$0.25_{10} \times 2_{10} = 0.5_{10}; \quad \lfloor 0.25_{10} \times 2_{10} \rfloor = \lfloor 0.5_{10} \rfloor = \mathbf{0}$$

$$0.5_{10} - \lfloor 0.5_{10} \rfloor = 0.5_{10}$$

$$0.5_{10} \times 2_{10} = 1.0_{10}; \quad \lfloor 1.0_{10} \rfloor = \mathbf{1}$$

# Trupmenų pavertimas

Trupmenas galime padauginti iš skaičiavimo sistemos bazės ir gauti vieną po kito skaitmenis už trupmenos kablelio:

Paverskime  $0.625_{10}$  į dvejetainę sistemą.

$$0.625_{10} \times 2_{10} = 1.25_{10}; \quad \lfloor 0.625_{10} \times 2_{10} \rfloor = \lfloor 1.25_{10} \rfloor = \mathbf{1}$$

$$1.25_{10} - \lfloor 1.25_{10} \rfloor = 0.25_{10}$$

$$0.25_{10} \times 2_{10} = 0.5_{10}; \quad \lfloor 0.25_{10} \times 2_{10} \rfloor = \lfloor 0.5_{10} \rfloor = \mathbf{0}$$

$$0.5_{10} - \lfloor 0.5_{10} \rfloor = 0.5_{10}$$

$$0.5_{10} \times 2_{10} = 1.0_{10}; \quad \lfloor 1.0_{10} \rfloor = \mathbf{1}$$

$$1.0_{10} - \lfloor 1.0_{10} \rfloor = 1 - 1 = 0 \Rightarrow \mathbf{end}$$

# Trupmenų pavertimas

Trupmenas galime padauginti iš skaičiavimo sistemos bazės ir gauti vieną po kito skaitmenis už trupmenos kablelio:

Paverskime  $0.625_{10}$  į dvejetainę sistemą.

$$0.625_{10} \times 2_{10} = 1.25_{10}; \quad \lfloor 0.625_{10} \times 2_{10} \rfloor = \lfloor 1.25_{10} \rfloor = \mathbf{1}$$

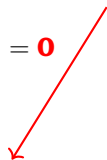
$$1.25_{10} - \lfloor 1.25_{10} \rfloor = 0.25_{10}$$

$$0.25_{10} \times 2_{10} = 0.5_{10}; \quad \lfloor 0.25_{10} \times 2_{10} \rfloor = \lfloor 0.5_{10} \rfloor = \mathbf{0}$$

$$0.5_{10} - \lfloor 0.5_{10} \rfloor = 0.5_{10}$$

$$0.5_{10} \times 2_{10} = 1.0_{10}; \quad \lfloor 1.0_{10} \rfloor = \mathbf{1}$$

$$1.0_{10} - \lfloor 1.0_{10} \rfloor = 1 - 1 = 0 \Rightarrow \mathbf{end}$$





# Trupmenų pavertimas

Trupmenas galime padauginti iš skaičiavimo sistemos bazės ir gauti vieną po kito skaitmenis už trupmenos kablelio:

Paverskime  $0.625_{10}$  į dvejetainę sistemą.

$$0.625_{10} \times 2_{10} = 1.25_{10}; \quad \lfloor 0.625_{10} \times 2_{10} \rfloor = \lfloor 1.25_{10} \rfloor = \mathbf{1}$$

$$1.25_{10} - \lfloor 1.25_{10} \rfloor = 0.25_{10}$$

$$0.25_{10} \times 2_{10} = 0.5_{10}; \quad \lfloor 0.25_{10} \times 2_{10} \rfloor = \lfloor 0.5_{10} \rfloor = \mathbf{0}$$

$$0.5_{10} - \lfloor 0.5_{10} \rfloor = 0.5_{10}$$

$$0.5_{10} \times 2_{10} = 1.0_{10}; \quad \lfloor 1.0_{10} \rfloor = \mathbf{1}$$

$$1.0_{10} - \lfloor 1.0_{10} \rfloor = 1 - 1 = 0 \Rightarrow \mathbf{end}$$

$0.101_2$

# Trupmenų pavertimas

Trupmenas galime padauginti iš skaičiavimo sistemos bazės ir gauti vieną po kito skaitmenis už trupmenos kablelio:

Paverskime  **$0.625_{10}$**  į dvejetainę sistemą.

$$0.625_{10} \times 2_{10} = 1.25_{10}; \quad \lfloor 0.625_{10} \times 2_{10} \rfloor = \lfloor 1.25_{10} \rfloor = \mathbf{1}$$

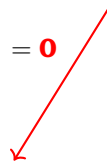
$$1.25_{10} - \lfloor 1.25_{10} \rfloor = 0.25_{10}$$

$$0.25_{10} \times 2_{10} = 0.5_{10}; \quad \lfloor 0.25_{10} \times 2_{10} \rfloor = \lfloor 0.5_{10} \rfloor = \mathbf{0}$$

$$0.5_{10} - \lfloor 0.5_{10} \rfloor = 0.5_{10}$$

$$0.5_{10} \times 2_{10} = 1.0_{10}; \quad \lfloor 1.0_{10} \rfloor = \mathbf{1}$$

$$1.0_{10} - \lfloor 1.0_{10} \rfloor = 1 - 1 = 0 \Rightarrow \mathbf{end}$$



$$0.101_2 = 1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3} = 0.5_{10} + 0.125_{10} = \mathbf{0.625_{10}}$$

# Pavertimas į aštuonetainę ir šešioliktainę sistemas

Dalyba iš  $8_{10}$  atitinka **3** dvejetainių skaitmenų atskyrimą:

$$1101\ 1011_2 / 8_{10} \equiv 1101\ 1011_2 / 1000_2 = 1101\ 1.\mathbf{011}_2$$

Dalyba iš  $16_{10}$  atitinka **4** dvejetainių skaitmenų atskyrimą:

$$1101\ 1011_2 / 16_{10} \equiv 1101\ 1011_2 / 10000_2 = 1101.\mathbf{1011}_2$$

⇒ Norint paversti į aštuonetainę (šešioliktainę) sistemą, reikia sugrupuoti dvejetainius skaitmenis po 3 (4) ir paversti kiekvieną grupę į atitinkamą kitos sistemos skaitmenį:

Aštuonetainiai			Šešioliktainiai skaitmenys			
000	0	←	0000	0	1000	8
001	1	Pavyzdys (aštuonetainis):	0001	1	1001	9
010	2	$01011011_2 = \mathbf{001\ 011\ 011}_2 = \mathbf{133}_8$	0010	2	1010	A
011	3		0011	3	1011	B
100	4		0100	4	1100	C
101	5		0101	5	1101	D
110	6	→	0110	6	1110	E
111	7	Pavyzdys (šešioliktainis):	0111	7	1111	F
		$01011011_2 = \mathbf{0101\ 1011}_2 = \mathbf{5B}_{16}$				

# XXI a.: pavertimas naudojant kompiuterį

Aprašymas:

```
man perlfaq4
```

Dvejetainė → dešimtainė:

```
perl -le 'print 0b_0110_1011'  
107
```

Šešioliktainė → dvejetainė:

```
perl -e 'printf "%016b\n", 0x23FF'  
0010001111111111
```

Aštuonetainė → šešioliktainė:

```
perl -e 'printf "%02X\n", 0133'  
5B
```

Dėmesio: šis metodas veikia tik mažiems skaičiams ( $|n| < 2^{31}$ !)

# XXI a.: trupmenų pavertimas

Paverskime trupmeną  $0.0110\ 1011_2 = 0110\ 1011_2 / 1\ 0000\ 0000_2$  į dešimtainę sistemą:

```
perl -le 'print 0b_0110_1011 / (2**8)'  
0.41796875
```

# XXI a.: trupmenų pavertimas

Paverskime trupmeną  $0.0110\ 1011_2 = 0110\ 1011_2 / 1\ 0000\ 0000_2$  į dešimtainę sistemą:

```
perl -le 'print 0b_0110_1011 / (2**8)'  
0.41796875
```

Patikrinkime:

```
perl -e '$x=0.41796875; print "0."; while($x) {$x*=2; print int($x); $x-=int($x)}; print "\n"  
0.01101011
```

Dėmesio! Šis metodas veikia **tik** tada, jeigu trupmenos **tiksliai** atvaizduojamos mūsų kompiuteryje!

$$0.1_{10} = 1/10_{10}$$

Failas: "examples/code/one-tenth.c"

```
#include <stdio.h>
#include <stdlib.h>

int main( int argc, char *argv[] )
{
    int n = 1; // numerator
    int d = 10; // denominator
    int b = 2; // the base of the target number system
    int s = 0; // current step
    int maxstep = 20;

    /* Get the max. digit count: */
    printf( "%d.", n/d );
    while( n != 0 && s < maxstep ) {
        n *= b;
        printf( "%d", n/d ); // |_ b*(n/d) _|
        n %= d; // b*(n/d) - |_ b*(n/d) _|
        s++;
        if( s % 4 == 0 ) {
            printf( " " ); // print space after each 4 digits
        }
    }
    printf( "\n" );

    return 0;
}
```

$$0.1_{10} = 1/10_{10}$$

```
sh$ examples/code/one-tenth
0.0001 1001 1001 1001 1001
```

$$0.1_{10} = 0.0001 \overline{1001}_2$$

Begalinė dvejetainė trupmena!

Jei pilnai suprastintos trupmenos vardiklis turi pirminių daugiklių, iš kurių **nesidalina** skaičių sistemos bazė, tokioje bazėje ši trupmena bus begalinė periodinė.

E.g.  $1/10_{10}$ :

$$\begin{aligned} 10_{10} &= 2 \times 5; \quad 5 \text{ nėra skaičiaus } 2 \text{ daliklis} \\ &\Rightarrow 1/10_{10} = 0.0001 \overline{1001}_2 \text{ - begalinė periodinė} \end{aligned}$$



# Dvejetainė aritmetika

*/.../ Būtinasis minimumas yra sudėtis, nes atimtis tai yra sudėtis su papildomu kodu, daugyba – daug sudėčių cikle, o dalyba – daug atimčių.*

Antanas Mitašiūnas (2016)

“Kompiuterių architektūra”, p. 9

Sudėtis:

$$A + B$$

			B	
	+	0	1	
	0	0	1	
A	1	1	10	



# Dvejetainė aritmetika

Sudėtis:

$$A + B$$

			B	
	+	0	1	
		0	1	
A		1	10	

Daugyba:

$$A \times B$$

			B	
	×	0	1	
		0	0	
A		1	01	

$$\begin{array}{r} 11101 \\ + 11001 \\ \hline \end{array}$$

# Dvejetainė aritmetika

Sudėtis:

$$A + B$$

$$\begin{array}{r} \phantom{A} \phantom{+} \phantom{|} \phantom{0} \phantom{1} \\ \phantom{A} \phantom{+} \phantom{|} \phantom{0} \phantom{1} \\ \phantom{A} \phantom{+} \phantom{|} \phantom{0} \phantom{1} \\ \hline A \phantom{+} \phantom{|} \phantom{0} \phantom{1} \\ 1 \phantom{|} \phantom{0} \phantom{1} \phantom{10} \end{array}$$

Daugyba:

$$A \times B$$

$$\begin{array}{r} \phantom{A} \phantom{\times} \phantom{|} \phantom{0} \phantom{1} \\ \phantom{A} \phantom{\times} \phantom{|} \phantom{0} \phantom{1} \\ \phantom{A} \phantom{\times} \phantom{|} \phantom{0} \phantom{1} \\ \hline A \phantom{\times} \phantom{|} \phantom{0} \phantom{1} \\ 1 \phantom{|} \phantom{0} \phantom{1} \end{array}$$

$$\begin{array}{r} \phantom{+} \phantom{1} \\ \phantom{+} \phantom{1} \\ \phantom{+} \phantom{1} \\ \phantom{+} \phantom{1} \\ \phantom{+} \phantom{1} \\ \hline + \phantom{1} \\ 11101 \\ 11001 \\ \hline 0 \end{array}$$



# Dvejetainė aritmetika

Sudėtis:

$$A + B$$

			B	
	+	0	1	
		0	1	
A		1	10	

Daugyba:

$$A \times B$$

			B	
	×	0	1	
		0	0	
A		1	01	

		1	
	+	11101	
		11001	
		110	



# Dvejetainė aritmetika

Sudėtis:

$$A + B$$

$$\begin{array}{r} \phantom{A} \phantom{+} \phantom{|} \phantom{0} \phantom{1} \\ \phantom{A} \phantom{+} \phantom{|} \phantom{0} \phantom{1} \\ \phantom{A} \phantom{+} \phantom{|} \phantom{0} \phantom{1} \\ \hline A \phantom{+} \phantom{|} \phantom{0} \phantom{1} \\ 1 \phantom{|} \phantom{0} \phantom{1} \phantom{10} \end{array}$$

Daugyba:

$$A \times B$$

$$\begin{array}{r} \phantom{A} \phantom{\times} \phantom{|} \phantom{0} \phantom{1} \\ \phantom{A} \phantom{\times} \phantom{|} \phantom{0} \phantom{1} \\ \phantom{A} \phantom{\times} \phantom{|} \phantom{0} \phantom{1} \\ \hline A \phantom{\times} \phantom{|} \phantom{0} \phantom{1} \\ 1 \phantom{|} \phantom{0} \phantom{1} \end{array}$$

$$\begin{array}{r} 11 \phantom{1} \\ + 11101 \\ + 11001 \\ \hline 10110 \end{array}$$



# Dvejetainė aritmetika

Sudėtis:

$$A + B$$

$$\begin{array}{r} \phantom{A} \phantom{+} \phantom{|} \phantom{0} \phantom{1} \\ \phantom{A} \phantom{+} \phantom{|} \phantom{0} \phantom{1} \\ \phantom{A} \phantom{+} \phantom{|} \phantom{0} \phantom{1} \\ \hline A \phantom{+} \phantom{|} \phantom{0} \phantom{1} \\ 1 \phantom{|} \phantom{0} \phantom{1} \phantom{10} \end{array}$$

Daugyba:

$$A \times B$$

$$\begin{array}{r} \phantom{A} \phantom{\times} \phantom{|} \phantom{0} \phantom{1} \\ \phantom{A} \phantom{\times} \phantom{|} \phantom{0} \phantom{1} \\ \phantom{A} \phantom{\times} \phantom{|} \phantom{0} \phantom{1} \\ \hline A \phantom{\times} \phantom{|} \phantom{0} \phantom{1} \\ 1 \phantom{|} \phantom{0} \phantom{1} \end{array}$$

$$\begin{array}{r} 11 \phantom{1} \\ + 11101 \\ + 11001 \\ \hline 110110 \end{array}$$



# Dvejetainė aritmetika

Sudėtis:

$$A + B$$

			B	
	+	0	1	
A	0	0	1	
	1	1	10	

	11	1	
	11101		
+	11001		
	110110		

Daugyba:

$$A \times B$$

			B	
	×	0	1	
A	0	0	0	
	1	0	1	

	1101	
×	101	
	1101	
	110110	

# Dvejetainė aritmetika

Sudėtis:

$$A + B$$

			B	
	+	0	1	
A	0	0	1	
	1	1	10	

$$\begin{array}{r} 11 \quad 1 \\ + 11101 \\ + 11001 \\ \hline 110110 \end{array}$$

Daugyba:

$$A \times B$$

			B	
	×	0	1	
A	0	0	0	
	1	0	1	

$$\begin{array}{r} \times \quad 1101 \\ \quad 101 \\ \hline \quad 1101 \\ \quad 0000 \\ \hline \end{array}$$

# Dvejetainė aritmetika

Sudėtis:

$$A + B$$

			B	
	+	0	1	
A	0	0	1	
	1	1	10	

	11	1	
	+	11101	
	+	11001	
		110110	

Daugyba:

$$A \times B$$

			B	
	×	0	1	
A	0	0	0	
	1	0	1	

	×	1101	
		101	
		1101	
		0000	
		1101	

# Dvejetainė aritmetika

Sudėtis:

$$A + B$$

			B	
	+	0	1	
A	0	0	1	
	1	1	10	

$$\begin{array}{r} 11 \quad 1 \\ + 11101 \\ + 11001 \\ \hline 110110 \end{array}$$

Daugyba:

$$A \times B$$

			B	
	×	0	1	
A	0	0	0	
	1	0	1	

$$\begin{array}{r} \times \quad 1101 \\ \quad 101 \\ \hline 111 \\ \quad 1101 \\ \quad \quad 0000 \\ \quad \quad 1101 \\ \hline 100001 \end{array}$$

# Loginės operacijos pabičiui

Vieną dvejetainę skiltį skaičiuje trumpai vadinsime *bitu*, iš anglų kalbos *binary digit*.

Naudingos loginės operacijos pabičiui:

Pasirinkti (maskuoti) bitus:

$$\begin{array}{r} 1101\ 1010\ 0111 \\ \text{AND } 0001\ 1111\ 1000 \\ \hline 0001\ 1010\ 0000 \end{array}$$

# Loginės operacijos pabičiui

Vieną dvejetainę skiltį skaičiuje trumpai vadinsime *bitu*, iš anglų kalbos *binary digit*.

Naudingos loginės operacijos pabičiui:

Pasirinkti (maskuoti) bitus:

$$\begin{array}{r} 1101\ 1010\ 0111 \\ \text{AND } 0001\ 1111\ 1000 \\ \hline 0001\ 1010\ 0000 \end{array}$$

Nustatyti bitus:

$$\begin{array}{r} 1101\ 1010\ 0111 \\ \text{OR } 0001\ 1111\ 1000 \\ \hline 1101\ 1111\ 1111 \end{array}$$



# Loginės operacijos pabičiui

Vieną dvejetainę skiltį skaičiuje trumpai vadinsime *bitu*, iš anglų kalbos *binary digit*.

Naudingos loginės operacijos pabičiui:

Pasirinkti (maskuoti) bitus:

$$\begin{array}{r} 1101\ 1010\ 0111 \\ \text{AND } 0001\ 1111\ 1000 \\ \hline 0001\ 1010\ 0000 \end{array}$$

Nustatyti bitus:

$$\begin{array}{r} 1101\ 1010\ 0111 \\ \text{OR } 0001\ 1111\ 1000 \\ \hline 1101\ 1111\ 1111 \end{array}$$

Klausimai:

$$2B \text{ OR NOT } 2B = ??$$

# Loginės operacijos pabičiui

Vieną dvejetainę skiltį skaičiuje trumpai vadinsime *bitu*, iš anglų kalbos *binary digit*.

Naudingos loginės operacijos pabičiui:

Pasirinkti (maskuoti) bitus:

$$\begin{array}{r} 1101\ 1010\ 0111 \\ \text{AND } 0001\ 1111\ 1000 \\ \hline 0001\ 1010\ 0000 \end{array}$$

Nustatyti bitus:

$$\begin{array}{r} 1101\ 1010\ 0111 \\ \text{OR } 0001\ 1111\ 1000 \\ \hline 1101\ 1111\ 1111 \end{array}$$

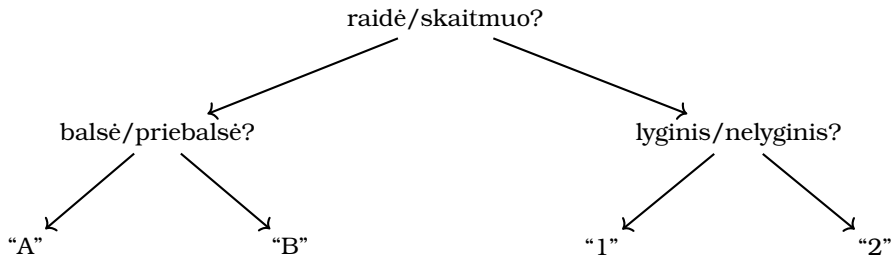
Klausimai:

2B OR NOT 2B = ??

Kokią bitų kaukę galima naudoti, norint patikrinti, ar skaičius yra lyginis?

# Intuityvus informacijos kiekio supratimas

Išsiaiškinti, kuris simbolis buvo perduotas, užduodant taip/ne klausimus: “A”, “B”, “1”, “2”.



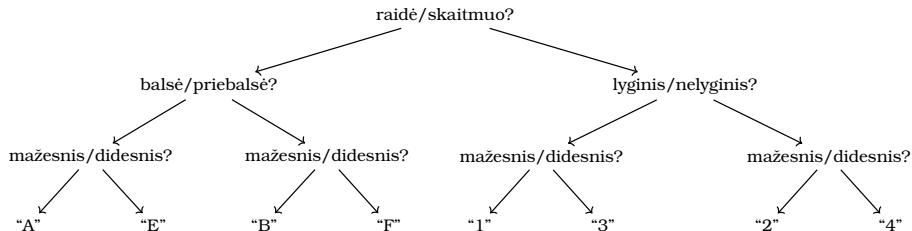
$$N_{\text{symbols}} = N_{\text{leaves}} = 2^{n_{\text{questions}}} = 2^{n_{\text{layers}}}$$

$$n_{\text{questions}} = n_{\text{layers}} = \log_2 N_{\text{symbols}} = \log_2 \frac{1}{p_{\text{symbol}}} = -\log_2 p_{\text{symbol}}$$

$$n_q = \log_2 N_s = \log_2 4 = -\log_2 \frac{1}{4} = \log_2 2^2 = 2$$

# Intuityvus informacijos kiekio supratimas (2)

Vienas iš 8 simbolių: “A”, “B”, “E”, “F”, “1”, “2”, “3”, “4”.



$$N_{\text{symbols}} = N_{\text{leaves}} = 2^{n_{\text{questions}}} = 2^{n_{\text{layers}}}$$

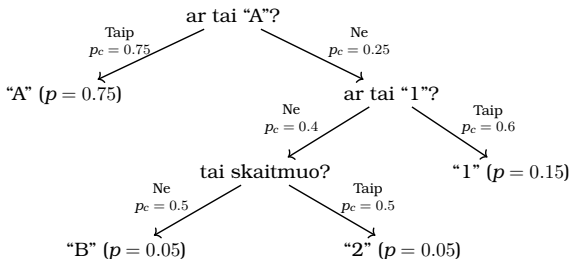
$$n_{\text{questions}} = n_{\text{layers}} = \log_2 N_{\text{symbols}} = \log_2 \frac{1}{p_{\text{symbol}}} = -\log_2 p_{\text{symbol}}$$

$$n_q = \log_2 N_s = \log_2 8 = -\log_2 \frac{1}{8} = \log_2 2^3 = 3$$

# Informacijos kiekis priklauso nuo tikimybių

Vienas iš 4 simbolių: “A”, “B”, “1”, “2”

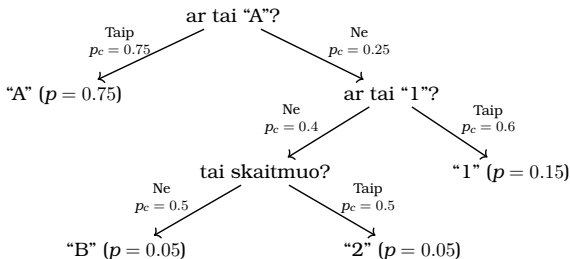
“AAA1AAABA1A2AAA1AAAA”  
(20 simbolių )



# Informacijos kiekis priklauso nuo tikimybių

Vienas iš 4 simbolių: “A”, “B”, “1”, “2”

“AAA1AAABA1A2AAAA1AAAA”  
(20 simbolių )

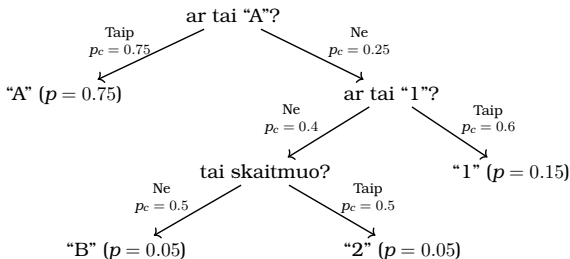


Esant vienodoms tikimybėms:  $n_q = \log_2 N_s = \log_2 \frac{1}{p_s} = -\log_2 p_s = \log_2 4 = \mathbf{2}$

# Informacijos kiekis priklauso nuo tikimybių

Vienas iš 4 simbolių: “A”, “B”, “1”, “2”

“AAA1AAABA1A2AAAA1AAAA”  
(20 simbolių )



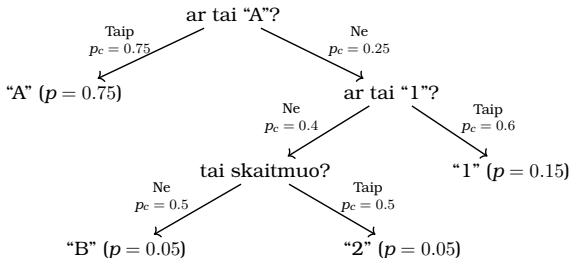
Esant vienodoms tikimybėms:  $n_q = \log_2 N_s = \log_2 \frac{1}{p_s} = -\log_2 p_s = \log_2 4 = \mathbf{2}$   
Pavaizduotiems klausimams (Hafmano kodas):

$$n_q = 0.75 + 2 \times 0.15 + 2 \times 3 \times 0.05 = \mathbf{1.35}$$

# Informacijos kiekis priklauso nuo tikimybių

Vienas iš 4 simbolių: “A”, “B”, “1”, “2”

“AAA1AAABA1A2AAAA1AAAA”  
(20 simbolių )



Esant vienodoms tikimybėms:  $n_q = \log_2 N_s = \log_2 \frac{1}{p_s} = -\log_2 p_s = \log_2 4 = \mathbf{2}$   
Pavaizduotiems klausimams (Hafmano kodas):

$$n_q = 0.75 + 2 \times 0.15 + 2 \times 3 \times 0.05 = \mathbf{1.35}$$

$$\text{Teorinė riba: } n_q = -\sum_{i=1}^4 p_i \log_2 p_i = \mathbf{1.15}$$



# Bitai ir baitai. Informacijos kiekis

Tarkime, kad yra šaltinis, siunčiantis pranešimus iš aibės  $S = \{M_1, \dots, M_n\}$ ,  $n \in \mathbb{N}$ .

Tada:

$$S = \{M_1, \dots, M_n\}$$

$$p_i = \Pr(M_i), \quad i = 1 \dots n$$

$$I_{M_i} = -\log p_i$$

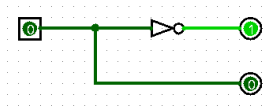
$$H = -\sum_{i=1}^n p_i \log p_i$$

Čia:

- $\Pr(M_i)$  – pranešimo  $M_i$  tikimybė;
- $I_{M_i}$  – informacijos kiekis pranešime  $M_i$ ;
- $H$  – šaltinio entropija, arba vidutinis informacijos kiekis pranešime;

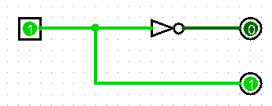
# Dviejų lygių signalas $\neq$ 1 bitas

Dviejų linijų vieno bito atvaizdavimas:



# Dviejų lygių signalas $\neq$ 1 bitas

Dviejų linijų vieno bito atvaizdavimas:



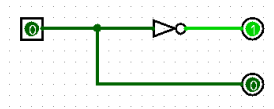
# Dviejų lygių signalas $\neq$ 1 bitas

$$\begin{aligned}Pr("0") &= Pr("1") = 0.5 \\ H &= -2 \cdot 0.5 \cdot \log_2 0.5 \\ &= -2 \cdot (-0.5) \\ &= \mathbf{1 \text{ bitas/pranešimui}}\end{aligned}$$

$$Pr("01") = Pr("10") = 0.5$$

$$Pr("00") = Pr("11") = 0$$

$$\begin{aligned}H &= -Pr("01") \cdot \log_2 Pr("01") \\ &\quad - Pr("10") \cdot \log_2 Pr("10") \\ &\quad - \lim_{Pr("00") \rightarrow 0} Pr("00") \cdot \log_2 Pr("00") \\ &\quad - \lim_{Pr("11") \rightarrow 0} Pr("11") \cdot \log_2 Pr("11") \\ &= -2 \cdot 0.5 \cdot (-1) \\ &= \mathbf{1 \text{ bitas/pranešimui}}\end{aligned}$$



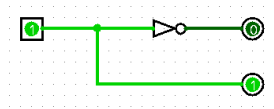
# Dviejų lygių signalas $\neq$ 1 bitas

$$\begin{aligned}Pr("0") &= Pr("1") = 0.5 \\H &= -2 \cdot 0.5 \cdot \log_2 0.5 \\&= -2 \cdot (-0.5) \\&= \mathbf{1 \text{ bitas/pranešimui}}\end{aligned}$$

$$Pr("01") = Pr("10") = 0.5$$

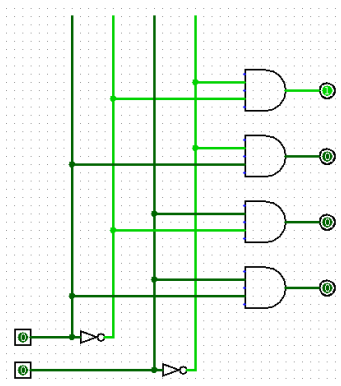
$$Pr("00") = Pr("11") = 0$$

$$\begin{aligned}H &= -Pr("01") \cdot \log_2 Pr("01") \\&\quad - Pr("10") \cdot \log_2 Pr("10") \\&\quad - \lim_{Pr("00") \rightarrow 0} Pr("00") \cdot \log_2 Pr("00") \\&\quad - \lim_{Pr("11") \rightarrow 0} Pr("11") \cdot \log_2 Pr("11") \\&= -2 \cdot 0.5 \cdot (-1) \\&= \mathbf{1 \text{ bitas/pranešimui}}\end{aligned}$$



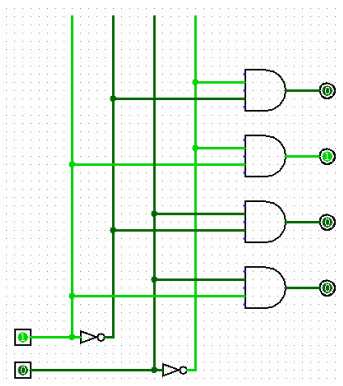
# Dviejų lygių signalas $\neq$ 1 bitas

## Dešifраторius 1-iš-4



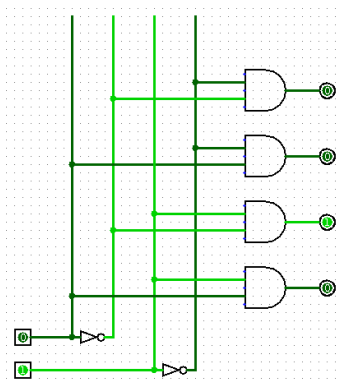
# Dviejų lygių signalas $\neq$ 1 bitas

## Dešifраторius 1-iš-4



# Dviejų lygių signalas $\neq$ 1 bitas

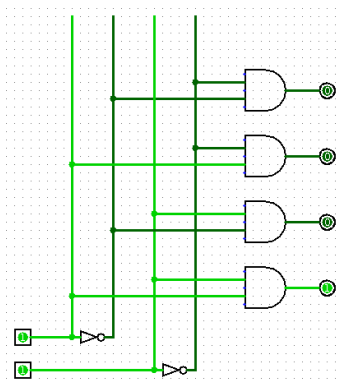
## Dešifраторius 1-iš-4





# Dviejų lygių signalas $\neq$ 1 bitas

## Dešifраторius 1-iš-4



# Pussumatoris

Sudėtis:				Suma:			Perkėlimas:		
$A + B$				<b>A</b>	<b>B</b>	<b>S</b>	<b>A</b>	<b>B</b>	<b>C</b>
			<b>B</b>	0	0	0	0	0	0
	+	0	1	0	1	1	0	1	0
		0	1	1	0	1	1	0	0
<b>A</b>		1	1	1	1	0	1	1	1

# Pussumatoris

Sudėtis:

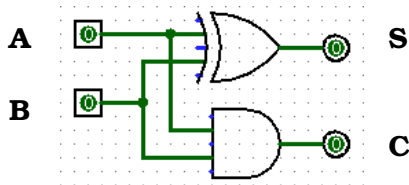
$$\begin{array}{r} A + B \\ \quad \quad \quad B \\ + \quad | \quad 0 \quad 1 \\ \hline 0 \quad | \quad 0 \quad 1 \\ 1 \quad | \quad 1 \quad 10 \end{array}$$

Suma:

A	B	S
0	0	0
0	1	1
1	0	1
1	1	0

Perkėlimas:

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1



# Pussumatoris

Sudėtis:

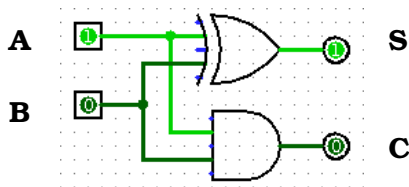
$$\begin{array}{r} A + B \\ \quad \quad \quad B \\ + \quad | \quad 0 \quad 1 \\ \hline 0 \quad | \quad 0 \quad 1 \\ 1 \quad | \quad 1 \quad 10 \end{array}$$

Suma:

<b>A</b>	<b>B</b>	<b>S</b>
0	0	0
0	1	1
1	0	1
1	1	0

Perkėlimas:

<b>A</b>	<b>B</b>	<b>C</b>
0	0	0
0	1	0
1	0	0
1	1	1



# Pussumatoris

Sudėtis:

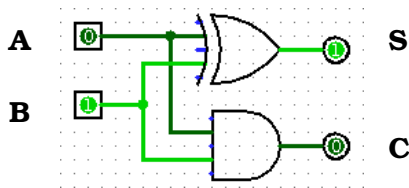
$$\begin{array}{r} A + B \\ \quad \quad \quad B \\ + \quad | \quad 0 \quad 1 \\ \hline 0 \quad | \quad 0 \quad 1 \\ 1 \quad | \quad 1 \quad 10 \end{array}$$

Suma:

<b>A</b>	<b>B</b>	<b>S</b>
0	0	0
0	1	1
1	0	1
1	1	0

Perkėlimas:

<b>A</b>	<b>B</b>	<b>C</b>
0	0	0
0	1	0
1	0	0
1	1	1





# Pilnas sumatorius

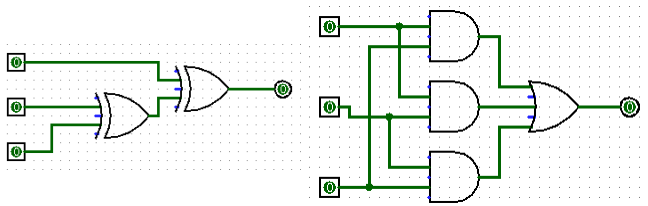
Suma:

$C_{in}$	A	B	S
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

Perkėlimas:

$C_{in}$	A	B	$C_{out}$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

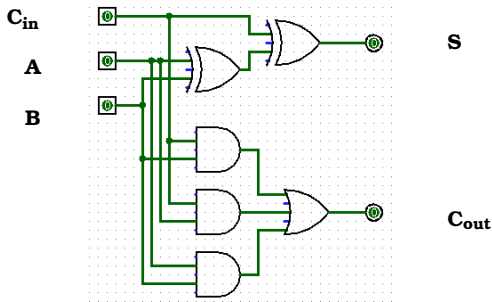
$\overline{C_{in}AB}$      $AB$   
 $C_{in}\overline{AB}$      $C_{in}B$   
 $C_{in}A\overline{B}$      $C_{in}A$   
 $C_{in}AB$



# Pilnas sumatorius (naivi realizacija)

Suma ir perkėlimas:

$C_{in}$	A	B	$C_{out}$	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

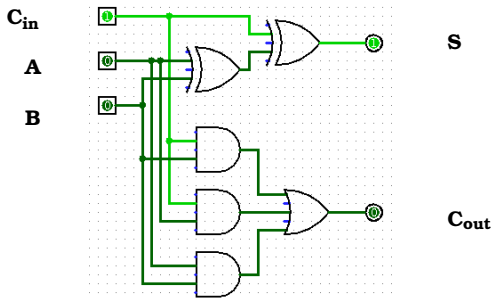




# Pilnas sumatorius (naivi realizacija)

Suma ir perkėlimas:

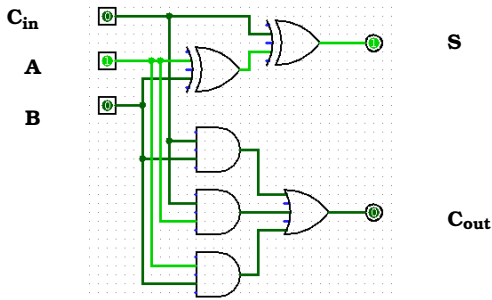
$C_{in}$	A	B	$C_{out}$	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



# Pilnas sumatorius (naivi realizacija)

Suma ir perkėlimas:

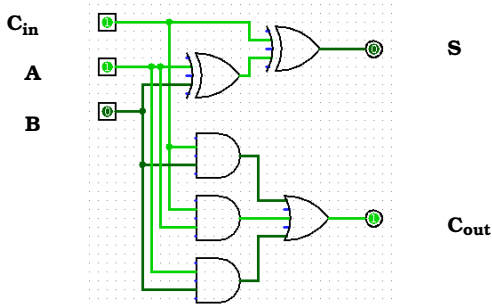
$C_{in}$	A	B	$C_{out}$	S
0	0	0	0	0
<b>0</b>	<b>0</b>	<b>1</b>	<b>0</b>	<b>1</b>
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



# Pilnas sumatorius (naivi realizacija)

Suma ir perkėlimas:

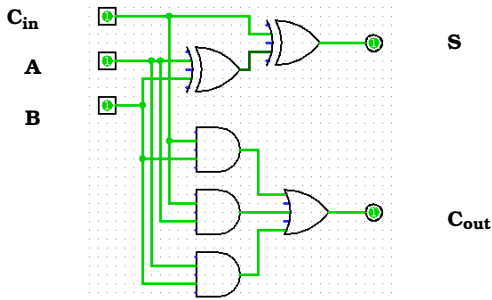
$C_{in}$	A	B	$C_{out}$	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
<b>1</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>0</b>
1	1	0	1	0
1	1	1	1	1



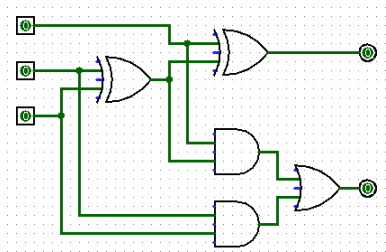
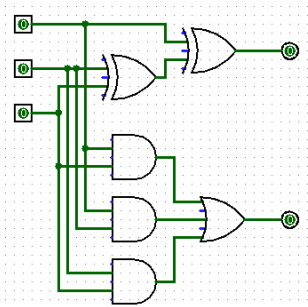
# Pilnas sumatorius (naivi realizacija)

Suma ir perkėlimas:

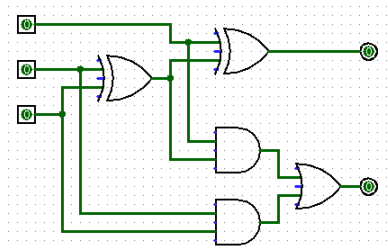
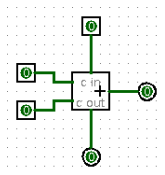
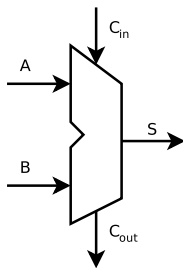
$C_{in}$	A	B	$C_{out}$	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



# Pilnas sumatorius

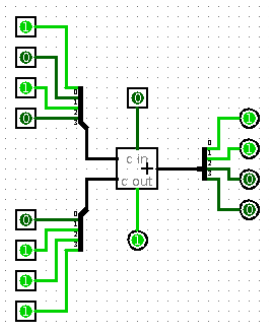
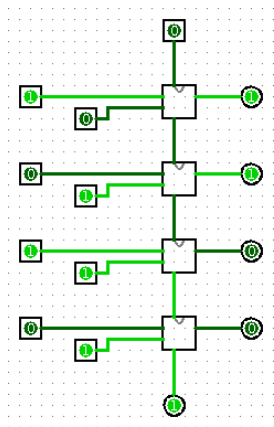


# Pilno sumatoriaus simboliai

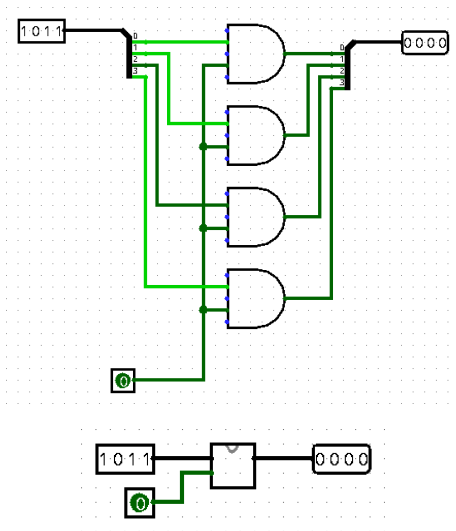


# 4 bitų sumatorius

$$0101_2 + 1110_2 = 10011_2 \quad 5_{16} + E_{16} = 13_{16} \quad 5_{10} + 14_{10} = 19_{10}$$

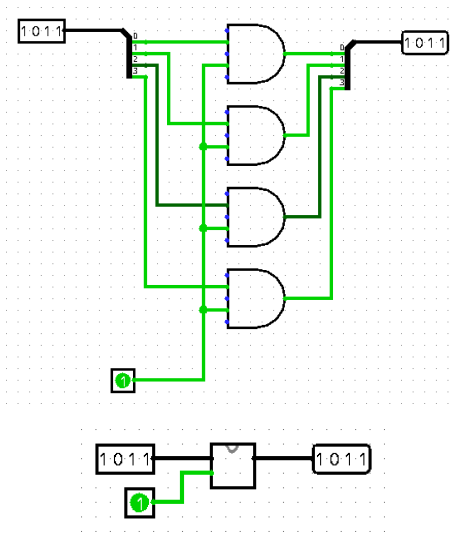


# Vienos skilties daugintuvas



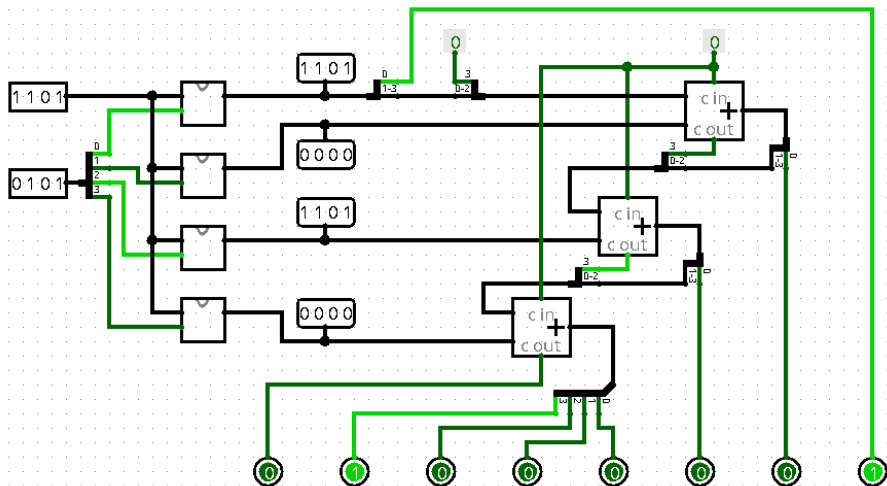


# Vienos skilties daugintuvas



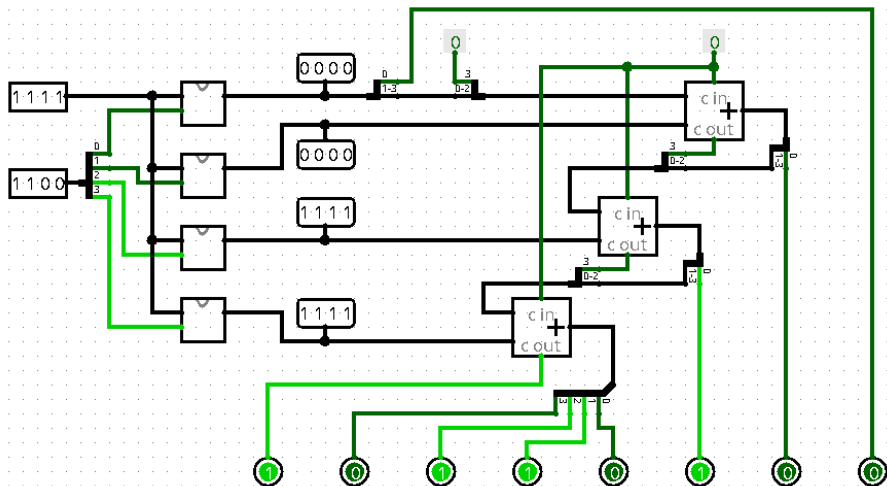
# 4 bitų daugintuvas

$$1101_2 \times 0101_2 = 0100\ 0001_2 \quad D_{16} \times 5_{16} = 41_{16} \quad 13_{10} \times 5_{10} = 65_{10}$$



# 4 bitų daugintuvas

$$1111_2 \times 1100_2 = 10110100_2 \quad F_{16} \times C_{16} = B_{416} \quad 15_{10} \times 12_{10} = 180_{10}$$



Atimti mažesnį iš didesnio lengva:

$$1101_2 - 100_2 = 1001_2$$

$$1101_2 - 11_2 = 11\overset{1}{0}1_2 - 11_2 = 1010_2$$

$$10_2 - 11_2 = ???$$

O kas bus, jei sudėsime?

$$\begin{array}{r} + \quad 0 \ 0 \ 0 \ 1 \\ \quad 1 \ 1 \ 1 \ 1 \\ \hline 1 \ 0 \ 0 \ 0 \ 0 \end{array}$$

Sudėkime didelius skaičius, kad atsirastų perkėlimas:

$$\begin{array}{r} + \quad \quad \quad 0 \ 0 \ 0 \ 1 \\ \quad \quad \quad 1 \ 1 \ 1 \ 1 \\ \hline 1 \ 0 \ 0 \ 0 \ 0 \end{array}$$

i.e.:

$$1111_2 + 1_2 = 0_2$$

bet mes žinome, kad

$$(-1) + 1 = 0$$

Ar galime pasakyti, kad  $1111_2 = -1_2$  tam tikra prasme?

# Papildinys iki dviejų

Pridėti 1111 reiškia:  $1111_2 = 1\ 0000_2 - 1 = 2^4 - 1$

Ignoruoti perkėlimo bitą – reiškia atimti  $1\ 0000_2 = 2^4$

Taigi

$$\begin{aligned} 1 + 1111_2 \text{ (ignoruojant perkėlimą)} &= 1 + (1\ 0000_2 - 1) - 1\ 0000_2 \\ &= 1 - 1 \\ &= 0 \end{aligned}$$

Patikrinkime – metodas veikia bet kokiam N bitų skaičiui:

$$a + \underbrace{(2^N - b)}_{b \text{ papildomasis kodas}} - 2^N = a - b$$

- Natūrali skaitmeninių kompiuterių skaičiavimo sistema – dvejetainė; yra paprasti algoritmai paversti skaičius į dvejetainę ir iš jos į kitas skaičiavimo sistemas, bet sistemos pagrindais 8 ir 16 yra pačios patogiausios.
- Perlo programavimo kalba galima lengvai paversti skaičius iš vienos sistemos į kitą;
- Ne kiekvieną skaičių galime paversti į kitą sistemą tiksliai!
- Ne visada signalas su dviem būsenom perduoda 1 bitą informacijos!
- Galime sukonstruoti logines grandines, kurios atlieka dvejetainę aritmetiką (sudėtį, atimtį, daugybą ir t.t.);

Mitašiūnas, Antanas (2016). *Kompiuterių architektūra*. URL:  
<http://www.mif.vu.lt/katedros/cs/Asmen/Kompiuteriu%20architektura.pdf>.