## Exercício 1:

**1. Decimal: 329**

Para converter o número decimal 329 para as outras bases, vamos usar as divisões sucessivas.

**Binário:**

Dividimos 329 por 2 e anotamos os restos.

* 329 ÷ 2 = 164, resto 1
* 164 ÷ 2 = 82, resto 0
* 82 ÷ 2 = 41, resto 0
* 41 ÷ 2 = 20, resto 1
* 20 ÷ 2 = 10, resto 0
* 10 ÷ 2 = 5, resto 0
* 5 ÷ 2 = 2, resto 1
* 2 ÷ 2 = 1, resto 0
* 1 ÷ 2 = 0, resto 1

Agora, pegamos os restos de baixo para cima: **101001001**.

Então, o número binário correspondente a 329 é **101001001**.

**Octal:**

Agora, agrupamos o número binário em grupos de 3 bits (da direita para a esquerda) e convertemos cada grupo para octal.

Binário: **101001001**

Agrupamos: **1 010 010 001** (adicionei zeros à esquerda para completar os grupos)

Agora, convertemos cada grupo:

* 001 (binário) = 1 (octal)
* 010 (binário) = 2 (octal)
* 010 (binário) = 2 (octal)
* 1 (binário) = 1 (octal)

Então, o número octal correspondente a 329 é **1221**.

**Hexadecimal:**

Agrupamos o número binário em grupos de 4 bits (da direita para a esquerda) e convertemos cada grupo para hexadecimal.

Binário: **101001001**

Agrupamos: **0001 0100 1001** (adicionei zeros à esquerda para completar os grupos)

Agora, convertemos cada grupo:

* 0001 (binário) = 1 (hexadecimal)
* 0100 (binário) = 4 (hexadecimal)
* 1001 (binário) = 9 (hexadecimal)

Então, o número hexadecimal correspondente a 329 é **149**.

**2. Binário: 11011101010**

Agora, vamos converter o número binário **11011101010** para as outras bases.

**Decimal:**

Para converter para decimal, multiplicamos os valores dos bits por 2 elevado à posição do bit, da direita para a esquerda.

110111010102​=1⋅210+1⋅29+0⋅28+1⋅27+1⋅26+1⋅25+0⋅24+1⋅23+0⋅22+1⋅21+0⋅20

Calculando:

1⋅1024+1⋅512+0⋅256+1⋅128+1⋅64+1⋅32+0⋅16+1⋅8+0⋅4+1⋅2+0⋅1=1106

Então, o número decimal correspondente a **11011101010** é **1106**.

**Octal:**

Primeiro, agrupamos o número binário em grupos de 3 bits (da direita para a esquerda).

**11011101010** → Agrupamos: **001 101 110 101 0** (adicionei zero à esquerda para completar o último grupo)

Agora, convertemos cada grupo para octal:

* 001 (binário) = 1 (octal)
* 101 (binário) = 5 (octal)
* 110 (binário) = 6 (octal)
* 101 (binário) = 5 (octal)
* 0 (binário) = 0 (octal)

Portanto, o número octal correspondente a **11011101010** é **15650**.

**Hexadecimal:**

Agora, agrupamos o número binário em grupos de 4 bits (da direita para a esquerda).

**11011101010** → Agrupamos: **0001 1011 1010 1010**

Agora, convertemos cada grupo para hexadecimal:

* 0001 (binário) = 1 (hexadecimal)
* 1011 (binário) = B (hexadecimal)
* 1010 (binário) = A (hexadecimal)
* 1010 (binário) = A (hexadecimal)

Então, o número hexadecimal correspondente a **11011101010** é **1BAA**.

**3. Octal: 1465**

Agora, vamos converter o número octal **1465** para as outras bases.

**Decimal:**

Para converter o número octal para decimal, multiplicamos cada dígito pelo valor da base elevada à sua posição (da direita para a esquerda):

14658​=1⋅83+4⋅82+6⋅81+5⋅80

Calculando:

1⋅512+4⋅64+6⋅8+5⋅1=512+256+48+5=821

Então, o número decimal correspondente a **1465** é **821**.

**Binário:**

Agora, convertemos o número octal para binário. Para isso, convertemos cada dígito octal para seu equivalente binário de 3 bits.

* 1 (octal) = 001 (binário)
* 4 (octal) = 100 (binário)
* 6 (octal) = 110 (binário)
* 5 (octal) = 101 (binário)

Portanto, o número binário correspondente a **1465** é **001100110101**.

**Hexadecimal:**

Agora, vamos converter o número octal **1465** para hexadecimal. Para isso, convertemos primeiro para binário e depois para hexadecimal.

Binário: **001100110101**

Agora, agrupamos em 4 bits e convertemos para hexadecimal:

* 0011 (binário) = 3 (hexadecimal)
* 0011 (binário) = 3 (hexadecimal)
* 0101 (binário) = 5 (hexadecimal)

Portanto, o número hexadecimal correspondente a **1465** é **335**.

**4. Hexadecimal: 33BD**

Por fim, vamos converter o número hexadecimal **33BD** para as outras bases.

**Decimal:**

Para converter hexadecimal para decimal, multiplicamos cada dígito pelo valor da base elevada à sua posição (da direita para a esquerda):

33BD16​=3⋅163+3⋅162+11⋅161+13⋅160

Calculando:

3⋅4096+3⋅256+11⋅16+13⋅1=12288+768+176+13=13045

Então, o número decimal correspondente a **33BD** é **13045**.

**Binário:**

Agora, convertemos o número hexadecimal para binário. Cada dígito hexadecimal é convertido para 4 bits:

* 3 (hexadecimal) = 0011 (binário)
* 3 (hexadecimal) = 0011 (binário)
* B (hexadecimal) = 1011 (binário)
* D (hexadecimal) = 1101 (binário)

Portanto, o número binário correspondente a **33BD** é **0011001110111101**.

**Octal:**

Agora, convertemos o número hexadecimal para octal. Primeiro, convertemos para binário:

Binário: **0011001110111101**

Agora, agrupamos em 3 bits e convertemos para octal:

* 001 (binário) = 1 (octal)
* 100 (binário) = 4 (octal)
* 111 (binário) = 7 (octal)
* 110 (binário) = 6 (octal)
* 101 (binário) = 5 (octal)

Portanto, o número octal correspondente a **33BD** é **14765**.

**Respostas finais:**

1. Decimal **329**:
   * Binário: **101001001**
   * Octal: **1221**
   * Hexadecimal: **149**
2. Binário **11011101010**:
   * Decimal: **1106**
   * Octal: **15650**
   * Hexadecimal: **1BAA**
3. Octal **1465**:
   * Decimal: **821**
   * Binário: **001100110101**
   * Hexadecimal: **335**
4. Hexadecimal **33BD**:
   * Decimal: **13045**
   * Binário: **0011001110111101**
   * Octal: **14765**

## Exercício 2:

Passo a passo:

1. Converter 45 para binário: 00101101

2. Como o número é negativo, o bit de sinal é 1.

3. Resultado final: 10101101

## Exercício 3:

1. Converter 584 para binário: 1001001000

2. Representar na forma normalizada: 1.001001000 × 2^9

3. Expoente com bias (9 + 15 = 24): 11000

4. Mantissa: 0010010000

Resultado final: 0 11000 0010010000

## Exercício 4:

## a) (A . B) + (A. 𝐵̅)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **A** | **B** | **𝐵̅** | **A . B** | **A . 𝐵̅** | **(A . B) + (A . 𝐵̅)** |
| 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 | 1 |

## b) (A + B) . (𝐴̅ + C)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **A** | **B** | **C** | **A̅** | **A + B** | **A̅ + C** | **(A + B) . (A̅ + C)** |
| 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 1 | 1 | 1 |

## c) A . (B + A̅ . C)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **A** | **B** | **C** | **A̅** | **A̅ . C** | **B + A̅ . C** | **A . (B + A̅ . C)** |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 0 | 1 | 1 |

## Exercício 5:

**Circuito (a)**

1. A primeira porta lógica é uma **AND** com entradas A e B:

X=A⋅B

1. A saída de **X** é conectada a uma porta **OR** junto com B:

S=X+B=(A⋅B)+B

1. Como B+(A⋅B)=B , a expressão final simplificada é:

S=B

**Circuito (b)**

1. Primeira **AND** tem três entradas A,B,C:

X=A⋅B⋅C

1. Segunda **AND** recebe A e C (C negado):

Y=A⋅C

1. Terceira **AND** recebe AAA e BBB:

S=X+Y+Z=(A⋅B⋅C)+(A⋅C)+(A⋅B)

**Circuito (c)**

1. Primeira **NOT** inverte A, gerando A
2. Segunda porta **OR** recebe A e B:

X=A+B

1. Segunda **AND** recebe C e B:

Y=B⋅C

1. Uma **AND** recebe X e Y, resultando em:

S=X⋅Y=(A+B)⋅(B⋅C)