_____ / _____ / _____

Armazenamento de flutuantes em máquinas

Em termos de armazenamento de números de ponto flutuante, uma máquina em particular é conhecida em função de 4 características que descrevem como o número é aproximado e depois registrado.

Uma máquina é descrita como

$$(\beta, t, [a, b])$$

sendo que cada símbolo significa

β é a base na qual os números são guardados. Esta informação não é armazenada em máquinas reais, já que neste caso este valor é fixo. Usualmente 2. Mas, em uma descrição teórica é a primeira coisa a se destacar.

t a quantidade de dígitos da mantissa do número. Ressalte-se que os números são guardados em modo normalizado, o que significa que não importa qual a sua magnitude, eles estarão no intervalo $(0, 1)$ ou seja, números fracionários com parte inteira igual a zero. É claro que nem o zero, nem o ponto são guardados. Assim o número 0.1234 terá mantissa igual a 1234, mas subentende-se que ele vale 0.1234. Se o número a registrar tiver mantissa mais comprida do que t dígitos, eles serão arredondados para caber em t dígitos.

a,b o intervalo aceito para o expoente do número. Lembrar que números que tenham exponente $e < a$ não serão guardados sinalizando-se **underflow** neste caso. Igualmente, números que tenham expoente $e > b$ também não serão registrados, sinalizando-se **overflow** neste caso.

Na prática

Vejam-se alguns exemplos. Seja a máquina

$$(6, 5, [-4, 4])$$

Como seria representado nela o número $124,45_{(10)}$?

- A primeira coisa a fazer é converter a parte inteira do número para a base pedida:
  $124 / 6 = 20$ e resto $= 4$
  $20 / 6 = 3$ e resto $= 2$
  $3 / 6 = 0$ e resto $= 3$
  com isso, a parte inteira $123_{(10)} = 324_{(6)}$

- Depois, deve-se converter a parte fracionária do número para a base pedida:
  $0.45 \times 6 = 2.7$ e o primeiro dígito é 2
  $0.7 \times 6 = 4.2$ e o segundo dígito é 4
  $0.2 \times 6 = 1.2$ e o terceiro dígito é 1
  $0.2 \times 6 = 1.2$ e o quarto dígito é 1

  e podemos parar por aqui, já que surge um dízima periódica
- Como resultado até aqui, tem-se $124,45_{(10)} = 324,241111..._{(6)}$
- Normalizar o número encontrado para achar o expoente devido. A idéia é mover a vírgula decimal até a posição anterior ao primeiro dígito. Então
  $324,241111..._{(6)} = 324,241111..._{(6)} \times 6^0$ e
  $324,241111..._{(6)} = 32,4241111..._{(6)} \times 6^1$ e
  $324,241111..._{(6)} = 3,24241111..._{(6)} \times 6^2$ e
  $324,241111..._{(6)} =, 324241111..._{(6)} \times 6^3$ e finalmente o número está normalizado.
- Testar se o expoente encontrado encaixa-se no intervalo definido $[a, b]$. Neste exemplo, deu-se $[-4, 4]$ e portanto o valor encontrado (3) pode ser registrado.
- Arredondar o valor encontrado com o tamanho definido na máquina: Como se viu $t = 5$ neste exemplo e portanto a mantissa a registrar é apenas 0.32424 com 5 dígitos. Para efeito de raciocínio, pode-se pensar neste número como $324,24$, já que ele continua com 5 dígitos e fica mais fácil pensar assim (eu acho).
- Erro absoluto: deve-se calcular agora qual o erro absoluto cometido. Para tanto, reconverta-se $324,24_{(6)}$ para a base decimal
  $3 \times 6^2 + 2 \times 6^1 + 4 \times 6^0 + 2 \times 6^{-1} + 4 \times 6^{-2}$
  $3 \times 36 + 2 \times 6 + 4 + 2 \times \frac{1}{6} + 4 \times \frac{1}{36}$
  $108 + 12 + 4 + 0.3333332 + 0.1111108 = 124.444444$
  O erro absoluto portanto é $|124,45 - 124.444444| = 0.0056$ (com 4 dígitos arredondados).
- Erro relativo: a divisão do erro absoluto pelo valor calculado ou: $\frac{0.0056}{124.44444} = 0.000045$ o que em 4 casas arredondadas dá 0.0000
- Para encerrar, o número seria representado nesta máquina como 0.32424×6^3

Exemplo 2 Seja agora representar o mesmo número $124,45_{(10)}$ na máquina $(7, 4, [-4, 4])$

- convertendo
  $124 / 7 = 17$ e resto $= 5$
  $17 / 7 = 2$ e resto $= 3$
  $2 / 7 = 0$ e resto 2

  $0.45 \times 7 = 3.15$
  $0.15 \times 7 = 1.05$
  $0.05 \times 7 = 0.35$
  $0.35 \times 7 = 2.45$ e podemos ir parando
  então $124,45_{(10)} = 235,3102_{(7)}$
- normalizar e achar o expoente
  $235,3102_7 = 235,3102_7 \times 7^0$
  $235,3102_7 = 23,53102_7 \times 7^1$
  $235,3102_7 = 2,353102_7 \times 7^2$
  $235,3102_7 =, 2353102_7 \times 7^3$
  o expoente é 3 e está dentro do limite.
- tamanho definido pela máquina ($t = 4$) e o número fica $0,2353$ ou $235,3$
- erro absoluto: reconverter $235,3_{(7)}$ para seu valor decimal $2 \times 7^2 + 3 \times 7^1 + 5 \times 7^0 + 3 \times 7^{-1}$
  $2 \times 49 + 3 \times 7 + 5 + 3 \times \frac{1}{7}$ $98 + 21 + 5 + 0.4285713 = 124,4285713$
- erro absoluto: $|124,45 - 124,4285713| = 0.02143$
- erro relativo: $\frac{0.02143}{124.42857} = 0.00017 \approx 0.0002$

Mais um exemplo Seja agora a máquina $(8, 3, [-4, 4])$ e vamos tentar representar nela o mesmo número $124,45_{(10)}$

- convertendo
  $124 / 8 = 15$ e resto $= 4$
  $15 / 8 = 1$ e resto $= 7$
  $1 / 8 = 0$ e resto 1

  $0.45 \times 8 = 3.6$
  $0.6 \times 8 = 4.8$
  $0.8 \times 8 = 6.4$
  $0.4 \times 8 = 3.2$ e podemos ir parando
  então $124,45_{(10)} = 174,3463_8$
- normalizar e achar o expoente
  $174,3463_8 = 174,3463_8 \times 8^0$
  $174,3463_8 = 17,43463_8 \times 8^1$
  $174,3463_8 = 1,743463_8 \times 8^2$
  $174,3463_8 =, 1743463_8 \times 8^3$
  o expoente é 3 e está dentro do limite.
- tamanho definido pela máquina ($t = 3$) e o número fica $0,174$ ou 174
- erro absoluto: reconverter 174_8 para seu valor decimal $1 \times 8^2 + 7 \times 8^1 + 4 \times 8^0$
  $64 + 56 + 4 = 124,0$

- erro absoluto: $|124,45 - 124| = 0.45$
- erro relativo: $\frac{0.45}{124} = 0.00362 \approx 0.0036$

Último exemplo Agora a máquina é $(9, 3, [-5, 5])$ e o número ainda é o mesmo $124,45_{(10)}$

- convertendo
  $124 / 9 = 13$ e resto $= 7$
  $13 / 9 = 1$ e resto $= 4$
  $1 / 9 = 0$ e resto 1

  $0.45 \times 9 = 4.05$
  $0.05 \times 9 = 0.45$
  $0.45 \times 9 = 4.05$ e podemos ir parando por conta da dízima
  então $124,45_{(10)} = 147,404040..._9$

- normalizar e achar o expoente
  $147,4040_9 = 147,4040_9 \times 9^0$
  $147,4040_9 = 14,74040_9 \times 9^1$
  $147,4040_9 = 1,474040_9 \times 9^2$
  $147,4040_9 =, 1474040_9 \times 9^3$
  o expoente é 3 e está dentro do limite.
- tamanho definido pela máquina ($t = 3$) e o número fica $0,147$ ou 147
- erro absoluto: reconverter 147_9 para seu valor decimal $1 \times 9^2 + 4 \times 9^1 + 7 \times 9^0$
  $81 + 36 + 7 = 124,0$
- erro absoluto: $|124,45 - 124| = 0.45$
- erro relativo: $\frac{0.45}{124} = 0.00362 \approx 0.0036$

Python

```
# maquina genérica
import numpy as np
def maq(beta,t,a,b,numero):
  pi=np.floor(numero)
  pf=numero-pi
  nb=[]
  while 1==1:
    nb=[pi%beta]+nb
    pi=pi//beta
    if pi==0:
      break
  fb=[]
  ct=0
  while ct<9:
    pf=pf*beta
    hora=np.floor(pf)
    fb=fb+[hora]
    pf=pf-hora
    ct=ct+1
  nu=nb+fb
  nu=nu[0:t]
  tamanho=len(nb)
  expoente=tamanho
  if expoente<a:
    print('underflow')
    return
  if expoente>b:
    print('overflow')
    return
  print('numero = 0.',end='')
  for i in range(len(nu)):
    print(int(nu[i]),end='')
  print(' * ',beta,' ^ ',expoente)
  bacana=0
  for i in range(len(nu)):
    bacana = bacana +
      (nu[i]*(beta**(tamanho-1)))
    tamanho=tamanho-1
  ea=np.around(np.abs(bacana-numero),4)
  er=np.around((ea/bacana),4)
  print('erro absoluto = ',ea)
  print('erro relativo = ',er)
  return

maq(6,5,-4,4,124.45)
```

com a resposta

```
numero = 0.32424 *  6  ^  3
erro absoluto =  0.0056
erro relativo =  0.0
```

☞ Para você fazer

A seguir, 4 instâncias do mesmo problema (armazenar um número e depois calcular o erro absoluto neste armazenamento.

	β	t	$[a$	$b]$	número a converter
1.	6	3	-8	8	4862.9
2.	8	5	-8	8	3058.64
3.	9	5	-8	8	64672.2
4.	7	4	-8	8	451.55

Responda aqui

	mantissa	erro absoluto
1		
2		
3		
4		

Armazenamento de flutuantes em máquinas

Em termos de armazenamento de números de ponto flutuante, uma máquina em particular é conhecida em função de 4 características que descrevem como o número é aproximado e depois registrado.

Uma máquina é descrita como

$$(\beta, t, [a, b])$$

sendo que cada símbolo significa

β é a base na qual os números são guardados. Esta informação não é armazenada em máquinas reais, já que neste caso este valor é fixo. Usualmente 2. Mas, em uma descrição teórica é a primeira coisa a se destacar.

t a quantidade de dígitos da mantissa do número. Ressalte-se que os números são guardados em modo normalizado, o que significa que não importa qual a sua magnitude, eles estarão no intervalo $(0, 1)$ ou seja, números fracionários com parte inteira igual a zero. É claro que nem o zero, nem o ponto são guardados. Assim o número 0.1234 terá mantissa igual a 1234, mas subentende-se que ele vale 0.1234. Se o número a registrar tiver mantissa mais comprida do que t dígitos, eles serão arredondados para caber em t dígitos.

a,b o intervalo aceito para o expoente do número. Lembrar que números que tenham exponente $e < a$ não serão guardados sinalizando-se **underflow** neste caso. Igualmente, números que tenham expoente $e > b$ também não serão registrados, sinalizando-se **overflow** neste caso.

Na prática

Vejam-se alguns exemplos. Seja a máquina

$$(6, 5, [-4, 4])$$

Como seria representado nela o número $124,45_{(10)}$?

- A primeira coisa a fazer é converter a parte inteira do número para a base pedida:
  $124 / 6 = 20$ e resto $= 4$
  $20 / 6 = 3$ e resto $= 2$
  $3 / 6 = 0$ e resto $= 3$
  com isso, a parte inteira $123_{(10)} = 324_{(6)}$

- Depois, deve-se converter a parte fracionária do número para a base pedida:
  $0.45 \times 6 = 2.7$ e o primeiro dígito é 2
  $0.7 \times 6 = 4.2$ e o segundo dígito é 4
  $0.2 \times 6 = 1.2$ e o terceiro dígito é 1
  $0.2 \times 6 = 1.2$ e o quarto dígito é 1

  e podemos parar por aqui, já que surge um dízima periódica
- Como resultado até aqui, tem-se $124,45_{(10)} = 324,241111\ldots_{(6)}$
- Normalizar o número encontrado para achar o expoente devido. A idéia é mover a vírgula decimal até a posição anterior ao primeiro dígito. Então
  $324,241111\ldots_{(6)} = 324,241111\ldots_{(6)} \times 6^0$ e
  $324,241111\ldots_{(6)} = 32,4241111\ldots_{(6)} \times 6^1$ e
  $324,241111\ldots_{(6)} = 3,24241111\ldots_{(6)} \times 6^2$ e
  $324,241111\ldots_{(6)} =, 324241111\ldots_{(6)} \times 6^3$ e finalmente o número está normalizado.
- Testar se o expoente encontrado encaixa-se no intervalo definido $[a, b]$. Neste exemplo, deu-se $[-4, 4]$ e portanto o valor encontrado (3) pode ser registrado.
- Arredondar o valor encontrado com o tamanho definido na máquina: Como se viu $t = 5$ neste exemplo e portanto a mantissa a registrar é apenas 0.32424 com 5 dígitos. Para efeito de raciocínio, pode-se pensar neste número como 324, 24, já que ele continua com 5 dígitos e fica mais fácil pensar assim (eu acho).
- Erro absoluto: deve-se calcular agora qual o erro absoluto cometido. Para tanto, reconverta-se $324,24_{(6)}$ para a base decimal
  $3 \times 6^2 + 2 \times 6^1 + 4 \times 6^0 + 2 \times 6^{-1} + 4 \times 6^{-2}$
  $3 \times 36 + 2 \times 6 + 4 + 2 \times \frac{1}{6} + 4 \times \frac{1}{36}$
  $108 + 12 + 4 + 0.3333332 + 0.1111108 = 124.444444$
  O erro absoluto portanto é $|124,45 - 124.444444| = 0.0056$ (com 4 dígitos arredondados).
- Erro relativo: a divisão do erro absoluto pelo valor calculado ou: $\frac{0.0056}{124.44444} = 0.000045$ o que em 4 casas arredondadas dá 0.0000
- Para encerrar, o número seria representado nesta máquina como 0.32424×6^3

Exemplo 2 Seja agora representar o mesmo número $124,45_{(10)}$ na máquina $(7, 4, [-4, 4])$

- convertendo
  $124 / 7 = 17$ e resto $= 5$
  $17 / 7 = 2$ e resto $= 3$
  $2 / 7 = 0$ e resto 2

  $0.45 \times 7 = 3.15$
  $0.15 \times 7 = 1.05$
  $0.05 \times 7 = 0.35$
  $0.35 \times 7 = 2.45$ e podemos ir parando
  então $124,45_{(10)} = 235,3102_{(7)}$
- normalizar e achar o expoente
  $235,3102_7 = 235,3102_7 \times 7^0$
  $235,3102_7 = 23,53102_7 \times 7^1$
  $235,3102_7 = 2,353102_7 \times 7^2$
  $235,3102_7 =, 2353102_7 \times 7^3$
  o expoente é 3 e está dentro do limite.
- tamanho definido pela máquina ($t = 4$) e o número fica $0,2353$ ou $235,3$
- erro absoluto: reconverter $235,3_{(7)}$ para seu valor decimal $2 \times 7^2 + 3 \times 7^1 + 5 \times 7^0 + 3 \times 7^{-1}$
  $2 \times 49 + 3 \times 7 + 5 + 3 \times \frac{1}{7}$ $98 + 21 + 5 + 0.4285713 = 124,4285713$
- erro absoluto: $|124,45 - 124,4285713| = 0.02143$
- erro relativo: $\frac{0.02143}{124.42857} = 0.00017 \approx 0.0002$

Mais um exemplo Seja agora a máquina $(8, 3, [-4, 4])$ e vamos tentar representar nela o mesmo número $124,45_{(10)}$

- convertendo
  $124 / 8 = 15$ e resto $= 4$
  $15 / 8 = 1$ e resto $= 7$
  $1 / 8 = 0$ e resto 1

  $0.45 \times 8 = 3.6$
  $0.6 \times 8 = 4.8$
  $0.8 \times 8 = 6.4$
  $0.4 \times 8 = 3.2$ e podemos ir parando
  então $124,45_{(10)} = 174,3463_8$
- normalizar e achar o expoente
  $174,3463_8 = 174,3463_8 \times 8^0$
  $174,3463_8 = 17,43463_8 \times 8^1$
  $174,3463_8 = 1,743463_8 \times 8^2$
  $174,3463_8 =, 1743463_8 \times 8^3$
  o expoente é 3 e está dentro do limite.
- tamanho definido pela máquina ($t = 3$) e o número fica $0,174$ ou 174
- erro absoluto: reconverter 174_8 para seu valor decimal $1 \times 8^2 + 7 \times 8^1 + 4 \times 8^0$
  $64 + 56 + 4 = 124,0$

- erro absoluto: $|124,45 - 124| = 0.45$
- erro relativo: $\frac{0.45}{124} = 0.00362 \approx 0.0036$

Último exemplo Agora a máquina é $(9, 3, [-5, 5])$ e o número ainda é o mesmo $124,45_{(10)}$

- convertendo
  $124 / 9 = 13$ e resto $= 7$
  $13 / 9 = 1$ e resto $= 4$
  $1 / 9 = 0$ e resto 1

  $0.45 \times 9 = 4.05$
  $0.05 \times 9 = 0.45$
  $0.45 \times 9 = 4.05$ e podemos ir parando por conta da dízima
  então $124,45_{(10)} = 147,404040\ldots_9$

- normalizar e achar o expoente
  $147,4040_9 = 147,4040_9 \times 9^0$
  $147,4040_9 = 14,74040_9 \times 9^1$
  $147,4040_9 = 1,474040_9 \times 9^2$
  $147,4040_9 =, 1474040_9 \times 9^3$
  o expoente é 3 e está dentro do limite.
- tamanho definido pela máquina ($t = 3$) e o número fica $0,147$ ou 147
- erro absoluto: reconverter 147_9 para seu valor decimal $1 \times 9^2 + 4 \times 9^1 + 7 \times 9^0$
  $81 + 36 + 7 = 124,0$
- erro absoluto: $|124,45 - 124| = 0.45$
- erro relativo: $\frac{0.45}{124} = 0.00362 \approx 0.0036$

Python

```
# maquina genérica
import numpy as np
def maq(beta,t,a,b,numero):
  pi=np.floor(numero)
  pf=numero-pi
  nb=[]
  while 1==1:
    nb=[pi%beta]+nb
    pi=pi//beta
    if pi==0:
      break
  fb=[]
  ct=0
  while ct<9:
    pf=pf*beta
    hora=np.floor(pf)
    fb=fb+[hora]
    pf=pf-hora
    ct=ct+1
  nu=nb+fb
  nu=nu[0:t]
  tamanho=len(nb)
  expoente=tamanho
  if expoente<a:
    print('underflow')
    return
  if expoente>b:
    print('overflow')
    return
  print('numero = 0.',end='')
  for i in range(len(nu)):
    print(int(nu[i]),end='')
  print(' * ',beta,' ^ ',expoente)
  bacana=0
  for i in range(len(nu)):
    bacana = bacana +
      (nu[i]*(beta**(tamanho-1)))
    tamanho=tamanho-1
  ea=np.around(np.abs(bacana-numero),4)
  er=np.around((ea/bacana),4)
  print('erro absoluto = ',ea)
  print('erro relativo = ',er)
  return

maq(6,5,-4,4,124.45)
```

com a resposta

```
numero = 0.32424 *  6  ^  3
erro absoluto =  0.0056
erro relativo =  0.0
```

☞ Para você fazer

A seguir, 4 instâncias do mesmo problema (armazenar um número e depois calcular o erro absoluto neste armazenamento.

	β	t	$[a$	$b]$	número a converter
1.	8	3	-7	7	1033.35
2.	6	5	-7	7	738.466
3.	7	4	-7	7	38857.8
4.	5	3	-7	7	821.927

Responda aqui

	mantissa	erro absoluto
1		
2		
3		
4		

Armazenamento de flutuantes em máquinas

Em termos de armazenamento de números de ponto flutuante, uma máquina em particular é conhecida em função de 4 características que descrevem como o número é aproximado e depois registrado.

Uma máquina é descrita como

$$(\beta, t, [a, b])$$

sendo que cada símbolo significa

β é a base na qual os números são guardados. Esta informação não é armazenada em máquinas reais, já que neste caso este valor é fixo. Usualmente 2. Mas, em uma descrição teórica é a primeira coisa a se destacar.

t a quantidade de dígitos da mantissa do número. Ressalte-se que os números são guardados em modo normalizado, o que significa que não importa qual a sua magnitude, eles estarão no intervalo $(0, 1)$ ou seja, números fracionários com parte inteira igual a zero. É claro que nem o zero, nem o ponto são guardados. Assim o número 0.1234 terá mantissa igual a 1234, mas subentende-se que ele vale 0.1234. Se o número a registrar tiver mantissa mais comprida do que t dígitos, eles serão arredondados para caber em t dígitos.

a,b o intervalo aceito para o expoente do número. Lembrar que números que tenham exponente $e < a$ não serão guardados sinalizando-se **underflow** neste caso. Igualmente, números que tenham expoente $e > b$ também não serão registrados, sinalizando-se **overflow** neste caso.

Na prática

Vejam-se alguns exemplos. Seja a máquina

$$(6, 5, [-4, 4])$$

Como seria representado nela o número $124,45_{(10)}$?

- A primeira coisa a fazer é converter a parte inteira do número para a base pedida:
  $124 / 6 = 20$ e resto $= 4$
  $20 / 6 = 3$ e resto $= 2$
  $3 / 6 = 0$ e resto $= 3$
  com isso, a parte inteira $123_{(10)} = 324_{(6)}$

- Depois, deve-se converter a parte fracionária do número para a base pedida:
  $0.45 \times 6 = 2.7$ e o primeiro dígito é 2
  $0.7 \times 6 = 4.2$ e o segundo dígito é 4
  $0.2 \times 6 = 1.2$ e o terceiro dígito é 1
  $0.2 \times 6 = 1.2$ e o quarto dígito é 1

  e podemos parar por aqui, já que surge um dízima periódica
- Como resultado até aqui, tem-se $124,45_{(10)} = 324,241111\ldots_{(6)}$
- Normalizar o número encontrado para achar o expoente devido. A idéia é mover a vírgula decimal até a posição anterior ao primeiro dígito. Então
  $324,241111\ldots_{(6)} = 324,241111\ldots_{(6)} \times 6^0$ e
  $324,241111\ldots_{(6)} = 32,4241111\ldots_{(6)} \times 6^1$ e
  $324,241111\ldots_{(6)} = 3,24241111\ldots_{(6)} \times 6^2$ e
  $324,241111\ldots_{(6)} =, 324241111\ldots_{(6)} \times 6^3$ e finalmente o número está normalizado.
- Testar se o expoente encontrado encaixa-se no intervalo definido $[a, b]$. Neste exemplo, deu-se $[-4, 4]$ e portanto o valor encontrado (3) pode ser registrado.
- Arredondar o valor encontrado com o tamanho definido na máquina: Como se viu $t = 5$ neste exemplo e portanto a mantissa a registrar é apenas 0.32424 com 5 dígitos. Para efeito de raciocínio, pode-se pensar neste número como 324, 24, já que ele continua com 5 dígitos e fica mais fácil pensar assim (eu acho).
- Erro absoluto: deve-se calcular agora qual o erro absoluto cometido. Para tanto, reconverta-se $324,24_{(6)}$ para a base decimal
  $3 \times 6^2 + 2 \times 6^1 + 4 \times 6^0 + 2 \times 6^{-1} + 4 \times 6^{-2}$
  $3 \times 36 + 2 \times 6 + 4 + 2 \times \frac{1}{6} + 4 \times \frac{1}{36}$
  $108 + 12 + 4 + 0.3333332 + 0.1111108 = 124.444444$
  O erro absoluto portanto é $|124,45 - 124.444444| = 0.0056$ (com 4 dígitos arredondados).
- Erro relativo: a divisão do erro absoluto pelo valor calculado ou: $\frac{0.0056}{124.44444} = 0.000045$ o que em 4 casas arredondadas dá 0.0000
- Para encerrar, o número seria representado nesta máquina como 0.32424×6^3

Exemplo 2 Seja agora representar o mesmo número $124,45_{(10)}$ na máquina $(7, 4, [-4, 4])$

- convertendo
  $124 / 7 = 17$ e resto $= 5$
  $17 / 7 = 2$ e resto $= 3$
  $2 / 7 = 0$ e resto 2

  $0.45 \times 7 = 3.15$
  $0.15 \times 7 = 1.05$
  $0.05 \times 7 = 0.35$
  $0.35 \times 7 = 2.45$ e podemos ir parando
  então $124,45_{(10)} = 235,3102_{(7)}$
- normalizar e achar o expoente
  $235,3102_7 = 235,3102_7 \times 7^0$
  $235,3102_7 = 23,53102_7 \times 7^1$
  $235,3102_7 = 2,353102_7 \times 7^2$
  $235,3102_7 =, 2353102_7 \times 7^3$
  o expoente é 3 e está dentro do limite.
- tamanho definido pela máquina ($t = 4$) e o número fica $0,2353$ ou $235,3$
- erro absoluto: reconverter $235,3_{(7)}$ para seu valor decimal $2 \times 7^2 + 3 \times 7^1 + 5 \times 7^0 + 3 \times 7^{-1}$
  $2 \times 49 + 3 \times 7 + 5 + 3 \times \frac{1}{7}$ $98 + 21 + 5 + 0.4285713 = 124,4285713$
- erro absoluto: $|124,45 - 124,4285713| = 0.02143$
- erro relativo: $\frac{0.02143}{124.42857} = 0.00017 \approx 0.0002$

Mais um exemplo Seja agora a máquina $(8, 3, [-4, 4])$ e vamos tentar representar nela o mesmo número $124,45_{(10)}$

- convertendo
  $124 / 8 = 15$ e resto $= 4$
  $15 / 8 = 1$ e resto $= 7$
  $1 / 8 = 0$ e resto 1

  $0.45 \times 8 = 3.6$
  $0.6 \times 8 = 4.8$
  $0.8 \times 8 = 6.4$
  $0.4 \times 8 = 3.2$ e podemos ir parando
  então $124,45_{(10)} = 174,3463_8$
- normalizar e achar o expoente
  $174,3463_8 = 174,3463_8 \times 8^0$
  $174,3463_8 = 17,43463_8 \times 8^1$
  $174,3463_8 = 1,743463_8 \times 8^2$
  $174,3463_8 =, 1743463_8 \times 8^3$
  o expoente é 3 e está dentro do limite.
- tamanho definido pela máquina ($t = 3$) e o número fica $0,174$ ou 174
- erro absoluto: reconverter 174_8 para seu valor decimal $1 \times 8^2 + 7 \times 8^1 + 4 \times 8^0$
  $64 + 56 + 4 = 124,0$

- erro absoluto: $|124,45 - 124| = 0.45$
- erro relativo: $\frac{0.45}{124} = 0.00362 \approx 0.0036$

Último exemplo Agora a máquina é $(9, 3, [-5, 5])$ e o número ainda é o mesmo $124,45_{(10)}$

- convertendo
  $124 / 9 = 13$ e resto $= 7$
  $13 / 9 = 1$ e resto $= 4$
  $1 / 9 = 0$ e resto 1

  $0.45 \times 9 = 4.05$
  $0.05 \times 9 = 0.45$
  $0.45 \times 9 = 4.05$ e podemos ir parando por conta da dízima
  então $124,45_{(10)} = 147,404040\ldots_9$

- normalizar e achar o expoente
  $147,4040_9 = 147,4040_9 \times 9^0$
  $147,4040_9 = 14,74040_9 \times 9^1$
  $147,4040_9 = 1,474040_9 \times 9^2$
  $147,4040_9 =, 1474040_9 \times 9^3$
  o expoente é 3 e está dentro do limite.
- tamanho definido pela máquina ($t = 3$) e o número fica $0,147$ ou 147
- erro absoluto: reconverter 147_9 para seu valor decimal $1 \times 9^2 + 4 \times 9^1 + 7 \times 9^0$
  $81 + 36 + 7 = 124,0$
- erro absoluto: $|124,45 - 124| = 0.45$
- erro relativo: $\frac{0.45}{124} = 0.00362 \approx 0.0036$

Python

```
# maquina genérica
import numpy as np
def maq(beta,t,a,b,numero):
  pi=np.floor(numero)
  pf=numero-pi
  nb=[]
  while 1==1:
    nb=[pi%beta]+nb
    pi=pi//beta
    if pi==0:
      break
  fb=[]
  ct=0
  while ct<9:
    pf=pf*beta
    hora=np.floor(pf)
    fb=fb+[hora]
    pf=pf-hora
    ct=ct+1
  nu=nb+fb
  nu=nu[0:t]
  tamanho=len(nb)
  expoente=tamanho
  if expoente<a:
    print('underflow')
    return
  if expoente>b:
    print('overflow')
    return
  print('numero = 0.',end='')
  for i in range(len(nu)):
    print(int(nu[i]),end='')
  print(' * ',beta,' ^ ',expoente)
  bacana=0
  for i in range(len(nu)):
    bacana = bacana +
      (nu[i]*(beta**(tamanho-1)))
    tamanho=tamanho-1
  ea=np.around(np.abs(bacana-numero),4)
  er=np.around((ea/bacana),4)
  print('erro absoluto = ',ea)
  print('erro relativo = ',er)
  return

maq(6,5,-4,4,124.45)
```

com a resposta

```
numero = 0.32424 *  6  ^  3
erro absoluto =  0.0056
erro relativo =  0.0
```

☞ Para você fazer

A seguir, 4 instâncias do mesmo problema (armazenar um número e depois calcular o erro absoluto neste armazenamento.

	β	t	$[a$	$b]$	número a converter
1.	9	5	-9	9	7750.08
2.	6	4	-9	9	54759.7
3.	8	6	-9	9	4585.48
4.	5	3	-9	9	811.24

Responda aqui

	mantissa	erro absoluto
1		
2		
3		
4		

_____ / _____ / _____

Armazenamento de flutuantes em máquinas

Em termos de armazenamento de números de ponto flutuante, uma máquina em particular é conhecida em função de 4 características que descrevem como o número é aproximado e depois registrado.

Uma máquina é descrita como

$$(\beta, t, [a, b])$$

sendo que cada símbolo significa

β é a base na qual os números são guardados. Esta informação não é armazenada em máquinas reais, já que neste caso este valor é fixo. Usualmente 2. Mas, em uma descrição teórica é a primeira coisa a se destacar.

t a quantidade de dígitos da mantissa do número. Ressalte-se que os números são guardados em modo normalizado, o que significa que não importa qual a sua magnitude, eles estarão no intervalo $(0, 1)$ ou seja, números fracionários com parte inteira igual a zero. É claro que nem o zero, nem o ponto são guardados. Assim o número 0.1234 terá mantissa igual a 1234, mas subentende-se que ele vale 0.1234. Se o número a registrar tiver mantissa mais comprida do que t dígitos, eles serão arredondados para caber em t dígitos.

a,b o intervalo aceito para o expoente do número. Lembrar que números que tenham exponente $e < a$ não serão guardados sinalizando-se **underflow** neste caso. Igualmente, números que tenham expoente $e > b$ também não serão registrados, sinalizando-se **overflow** neste caso.

Na prática

Vejam-se alguns exemplos. Seja a máquina

$$(6, 5, [-4, 4])$$

Como seria representado nela o número $124,45_{(10)}$?

- A primeira coisa a fazer é converter a parte inteira do número para a base pedida:
  124 / 6 = 20 e resto = 4
  20 / 6 = 3 e resto = 2
  3 / 6 = 0 e resto = 3
  com isso, a parte inteira $123_{(10)} = 324_{(6)}$

- Depois, deve-se converter a parte fracionária do número para a base pedida:
  $0.45 \times 6 = 2.7$ e o primeiro dígito é 2
  $0.7 \times 6 = 4.2$ e o segundo dígito é 4
  $0.2 \times 6 = 1.2$ e o terceiro dígito é 1
  $0.2 \times 6 = 1.2$ e o quarto dígito é 1

  e podemos parar por aqui, já que surge um dízima periódica
- Como resultado até aqui, tem-se $124,45_{(10)} = 324,241111..._{(6)}$
- Normalizar o número encontrado para achar o expoente devido. A idéia é mover a vírgula decimal até a posição anterior ao primeiro dígito. Então
  $324,241111..._{(6)} = 324,241111..._{(6)} \times 6^0$ e
  $324,241111..._{(6)} = 32,4241111..._{(6)} \times 6^1$ e
  $324,241111..._{(6)} = 3,24241111..._{(6)} \times 6^2$ e
  $324,241111..._{(6)} =, 324241111..._{(6)} \times 6^3$ e finalmente o número está normalizado.
- Testar se o expoente encontrado encaixa-se no intervalo definido $[a, b]$. Neste exemplo, deu-se $[-4, 4]$ e portanto o valor encontrado (3) pode ser registrado.
- Arredondar o valor encontrado com o tamanho definido na máquina: Como se viu $t = 5$ neste exemplo e portanto a mantissa a registrar é apenas 0.32424 com 5 dígitos. Para efeito de raciocínio, pode-se pensar neste número como 324, 24, já que ele continua com 5 dígitos e fica mais fácil pensar assim (eu acho).
- Erro absoluto: deve-se calcular agora qual o erro absoluto cometido. Para tanto, reconverta-se $324,24_{(6)}$ para a base decimal
  $3 \times 6^2 + 2 \times 6^1 + 4 \times 6^0 + 2 \times 6^{-1} + 4 \times 6^{-2}$
  $3 \times 36 + 2 \times 6 + 4 + 2 \times \frac{1}{6} + 4 \times \frac{1}{36}$
  $108 + 12 + 4 + 0.3333332 + 0.1111108 = 124.444444$
  O erro absoluto portanto é $|124,45 - 124.444444| = 0.0056$ (com 4 dígitos arredondados).
- Erro relativo: a divisão do erro absoluto pelo valor calculado ou: $\frac{0.0056}{124.44444} = 0.000045$ o que em 4 casas arredondadas dá 0.0000
- Para encerrar, o número seria representado nesta máquina como 0.32424×6^3

Exemplo 2 Seja agora representar o mesmo número $124,45_{(10)}$ na máquina $(7, 4, [-4, 4])$

- convertendo
  124 / 7 = 17 e resto = 5
  17 / 7 = 2 e resto = 3
  2 / 7 = 0 e resto 2

  $0.45 \times 7 = 3.15$
  $0.15 \times 7 = 1.05$
  $0.05 \times 7 = 0.35$
  $0.35 \times 7 = 2.45$ e podemos ir parando
  então $124,45_{(10)} = 235,3102_{(7)}$
- normalizar e achar o expoente
  $235,3102_7 = 235,3102_7 \times 7^0$
  $235,3102_7 = 23,53102_7 \times 7^1$
  $235,3102_7 = 2,353102_7 \times 7^2$
  $235,3102_7 =, 2353102_7 \times 7^3$
  o expoente é 3 e está dentro do limite.
- tamanho definido pela máquina ($t = 4$) e o número fica $0,2353$ ou $235,3$
- erro absoluto: reconverter $235,3_{(7)}$ para seu valor decimal $2 \times 7^2 + 3 \times 7^1 + 5 \times 7^0 + 3 \times 7^{-1}$
  $2 \times 49 + 3 \times 7 + 5 + 3 \times \frac{1}{7}$ $98 + 21 + 5 + 0.4285713 = 124,4285713$
- erro absoluto: $|124,45 - 124,4285713| = 0.02143$
- erro relativo: $\frac{0.02143}{124.42857} = 0.00017 \approx 0.0002$

Mais um exemplo Seja agora a máquina $(8, 3, [-4, 4])$ e vamos tentar representar nela o mesmo número $124,45_{(10)}$

- convertendo
  124 / 8 = 15 e resto = 4
  15 / 8 = 1 e resto = 7
  1 / 8 = 0 e resto 1

  $0.45 \times 8 = 3.6$
  $0.6 \times 8 = 4.8$
  $0.8 \times 8 = 6.4$
  $0.4 \times 8 = 3.2$ e podemos ir parando
  então $124,45_{(10)} = 174,3463_8$
- normalizar e achar o expoente
  $174,3463_8 = 174,3463_8 \times 8^0$
  $174,3463_8 = 17,43463_8 \times 8^1$
  $174,3463_8 = 1,743463_8 \times 8^2$
  $174,3463_8 =, 1743463_8 \times 8^3$
  o expoente é 3 e está dentro do limite.
- tamanho definido pela máquina ($t = 3$) e o número fica $0,174$ ou 174
- erro absoluto: reconverter 174_8 para seu valor decimal $1 \times 8^2 + 7 \times 8^1 + 4 \times 8^0$
  $64 + 56 + 4 = 124,0$

- erro absoluto: $|124,45 - 124| = 0.45$
- erro relativo: $\frac{0.45}{124} = 0.00362 \approx 0.0036$

Último exemplo Agora a máquina é $(9, 3, [-5, 5])$ e o número ainda é o mesmo $124,45_{(10)}$

- convertendo
  124 / 9 = 13 e resto = 7
  13 / 9 = 1 e resto = 4
  1 / 9 = 0 e resto 1

  $0.45 \times 9 = 4.05$
  $0.05 \times 9 = 0.45$
  $0.45 \times 9 = 4.05$ e podemos ir parando por conta da dízima
  então $124,45_{(10)} = 147,404040..._9$

- normalizar e achar o expoente
  $147,4040_9 = 147,4040_9 \times 9^0$
  $147,4040_9 = 14,74040_9 \times 9^1$
  $147,4040_9 = 1,474040_9 \times 9^2$
  $147,4040_9 =, 1474040_9 \times 9^3$
  o expoente é 3 e está dentro do limite.
- tamanho definido pela máquina ($t = 3$) e o número fica $0,147$ ou 147
- erro absoluto: reconverter 147_9 para seu valor decimal $1 \times 9^2 + 4 \times 9^1 + 7 \times 9^0$
  $81 + 36 + 7 = 124,0$
- erro absoluto: $|124,45 - 124| = 0.45$
- erro relativo: $\frac{0.45}{124} = 0.00362 \approx 0.0036$

Python

```
# maquina genérica
import numpy as np
def maq(beta,t,a,b,numero):
  pi=np.floor(numero)
  pf=numero-pi
  nb=[]
  while 1==1:
    nb=[pi%beta]+nb
    pi=pi//beta
    if pi==0:
      break
  fb=[]
  ct=0
  while ct<9:
    pf=pf*beta
    hora=np.floor(pf)
    fb=fb+[hora]
    pf=pf-hora
    ct=ct+1
  nu=nb+fb
  nu=nu[0:t]
  tamanho=len(nb)
  expoente=tamanho
  if expoente<a:
    print('underflow')
    return
  if expoente>b:
    print('overflow')
    return
  print('numero = 0.',end='')
  for i in range(len(nu)):
    print(int(nu[i]),end='')
  print(' * ',beta,' ^ ',expoente)
  bacana=0
  for i in range(len(nu)):
    bacana = bacana +
      (nu[i]*(beta**(tamanho-1)))
    tamanho=tamanho-1
  ea=np.around(np.abs(bacana-numero),4)
  er=np.around((ea/bacana),4)
  print('erro absoluto = ',ea)
  print('erro relativo = ',er)
  return

maq(6,5,-4,4,124.45)
```

com a resposta

```
numero = 0.32424 *  6  ^  3
erro absoluto =  0.0056
erro relativo =  0.0
```

☞ Para você fazer

A seguir, 4 instâncias do mesmo problema (armazenar um número e depois calcular o erro absoluto neste armazenamento.

	β	t	$[a$	$b]$	número a converter
1.	7	6	-7	7	694.864
2.	9	5	-7	7	563.816
3.	5	6	-7	7	432.18
4.	8	3	-7	7	3552.43

Responda aqui

	mantissa	erro absoluto
1		
2		
3		
4		

UFPR - UP - UTFPR - PUCPr – 11/06/2022 - 13:26:59.6
Prof Dr P Kantek (pkantek@gmail.com)
Armazenamento de flutuantes em máquinas
VIVOk08a, V: 1.05 5
- 5 _____ / _____ / _____

Armazenamento de flutuantes em máquinas

Em termos de armazenamento de números de ponto flutuante, uma máquina em particular é conhecida em função de 4 características que descrevem como o número é aproximado e depois registrado.

Uma máquina é descrita como

$$(\beta, t, [a, b])$$

sendo que cada símbolo significa

β é a base na qual os números são guardados. Esta informação não é armazenada em máquinas reais, já que neste caso este valor é fixo. Usualmente 2. Mas, em uma descrição teórica é a primeira coisa a se destacar.

t a quantidade de dígitos da mantissa do número. Ressalte-se que os números são guardados em modo normalizado, o que significa que não importa qual a sua magnitude, eles estarão no intervalo $(0, 1)$ ou seja, números fracionários com parte inteira igual a zero. É claro que nem o zero, nem o ponto são guardados. Assim o número 0.1234 terá mantissa igual a 1234, mas subentende-se que ele vale 0.1234. Se o número a registrar tiver mantissa mais comprida do que t dígitos, eles serão arredondados para caber em t dígitos.

a,b o intervalo aceito para o expoente do número. Lembrar que números que tenham exponente $e < a$ não serão guardados sinalizando-se **underflow** neste caso. Igualmente, números que tenham expoente $e > b$ também não serão registrados, sinalizando-se **overflow** neste caso.

Na prática

Vejam-se alguns exemplos. Seja a máquina

$$(6, 5, [-4, 4])$$

Como seria representado nela o número $124,45_{(10)}$?

- A primeira coisa a fazer é converter a parte inteira do número para a base pedida:
  $124 / 6 = 20$ e resto $= 4$
  $20 / 6 = 3$ e resto $= 2$
  $3 / 6 = 0$ e resto $= 3$
  com isso, a parte inteira $123_{(10)} = 324_{(6)}$

- Depois, deve-se converter a parte fracionária do número para a base pedida:
  $0.45 \times 6 = 2.7$ e o primeiro dígito é 2
  $0.7 \times 6 = 4.2$ e o segundo dígito é 4
  $0.2 \times 6 = 1.2$ e o terceiro dígito é 1
  $0.2 \times 6 = 1.2$ e o quarto dígito é 1

  e podemos parar por aqui, já que surge um dízima periódica
- Como resultado até aqui, tem-se $124,45_{(10)} = 324,241111\ldots_{(6)}$
- Normalizar o número encontrado para achar o expoente devido. A idéia é mover a vírgula decimal até a posição anterior ao primeiro dígito. Então
  $324,241111\ldots_{(6)} = 324,241111\ldots_{(6)} \times 6^0$ e
  $324,241111\ldots_{(6)} = 32,4241111\ldots_{(6)} \times 6^1$ e
  $324,241111\ldots_{(6)} = 3,24241111\ldots_{(6)} \times 6^2$ e
  $324,241111\ldots_{(6)} =, 324241111\ldots_{(6)} \times 6^3$ e finalmente o número está normalizado.
- Testar se o expoente encontrado encaixa-se no intervalo definido $[a, b]$. Neste exemplo, deu-se $[-4, 4]$ e portanto o valor encontrado (3) pode ser registrado.
- Arredondar o valor encontrado com o tamanho definido na máquina: Como se viu $t = 5$ neste exemplo e portanto a mantissa a registrar é apenas 0.32424 com 5 dígitos. Para efeito de raciocínio, pode-se pensar neste número como 324, 24, já que ele continua com 5 dígitos e fica mais fácil pensar assim (eu acho).
- Erro absoluto: deve-se calcular agora qual o erro absoluto cometido. Para tanto, reconverta-se $324,24_{(6)}$ para a base decimal
  $3 \times 6^2 + 2 \times 6^1 + 4 \times 6^0 + 2 \times 6^{-1} + 4 \times 6^{-2}$
  $3 \times 36 + 2 \times 6 + 4 + 2 \times \frac{1}{6} + 4 \times \frac{1}{36}$
  $108 + 12 + 4 + 0.3333332 + 0.1111108 = 124.444444$
  O erro absoluto portanto é $|124,45 - 124.444444| = 0.0056$ (com 4 dígitos arredondados).
- Erro relativo: a divisão do erro absoluto pelo valor calculado ou: $\frac{0.0056}{124.44444} = 0.000045$ o que em 4 casas arredondadas dá 0.0000
- Para encerrar, o número seria representado nesta máquina como 0.32424×6^3

Exemplo 2 Seja agora representar o mesmo número $124,45_{(10)}$ na máquina $(7, 4, [-4, 4])$

- convertendo
  $124 / 7 = 17$ e resto $= 5$
  $17 / 7 = 2$ e resto $= 3$
  $2 / 7 = 0$ e resto 2

  $0.45 \times 7 = 3.15$
  $0.15 \times 7 = 1.05$
  $0.05 \times 7 = 0.35$
  $0.35 \times 7 = 2.45$ e podemos ir parando
  então $124,45_{(10)} = 235,3102_{(7)}$
- normalizar e achar o expoente
  $235,3102_7 = 235,3102_7 \times 7^0$
  $235,3102_7 = 23,53102_7 \times 7^1$
  $235,3102_7 = 2,353102_7 \times 7^2$
  $235,3102_7 =, 2353102_7 \times 7^3$
  o expoente é 3 e está dentro do limite.
- tamanho definido pela máquina ($t = 4$) e o número fica $0,2353$ ou $235,3$
- erro absoluto: reconverter $235,3_{(7)}$ para seu valor decimal $2 \times 7^2 + 3 \times 7^1 + 5 \times 7^0 + 3 \times 7^{-1}$
  $2 \times 49 + 3 \times 7 + 5 + 3 \times \frac{1}{7}$ $98 + 21 + 5 + 0.4285713 = 124,4285713$
- erro absoluto: $|124,45 - 124,4285713| = 0.02143$
- erro relativo: $\frac{0.02143}{124.42857} = 0.00017 \approx 0.0002$

Mais um exemplo Seja agora a máquina $(8, 3, [-4, 4])$ e vamos tentar representar nela o mesmo número $124,45_{(10)}$

- convertendo
  $124 / 8 = 15$ e resto $= 4$
  $15 / 8 = 1$ e resto $= 7$
  $1 / 8 = 0$ e resto 1

  $0.45 \times 8 = 3.6$
  $0.6 \times 8 = 4.8$
  $0.8 \times 8 = 6.4$
  $0.4 \times 8 = 3.2$ e podemos ir parando
  então $124,45_{(10)} = 174,3463_8$
- normalizar e achar o expoente
  $174,3463_8 = 174,3463_8 \times 8^0$
  $174,3463_8 = 17,43463_8 \times 8^1$
  $174,3463_8 = 1,743463_8 \times 8^2$
  $174,3463_8 =, 1743463_8 \times 8^3$
  o expoente é 3 e está dentro do limite.
- tamanho definido pela máquina ($t = 3$) e o número fica $0,174$ ou 174
- erro absoluto: reconverter 174_8 para seu valor decimal $1 \times 8^2 + 7 \times 8^1 + 4 \times 8^0$
  $64 + 56 + 4 = 124,0$

- erro absoluto: $|124,45 - 124| = 0.45$
- erro relativo: $\frac{0.45}{124} = 0.00362 \approx 0.0036$

Último exemplo Agora a máquina é $(9, 3, [-5, 5])$ e o número ainda é o mesmo $124,45_{(10)}$

- convertendo
  $124 / 9 = 13$ e resto $= 7$
  $13 / 9 = 1$ e resto $= 4$
  $1 / 9 = 0$ e resto 1

  $0.45 \times 9 = 4.05$
  $0.05 \times 9 = 0.45$
  $0.45 \times 9 = 4.05$ e podemos ir parando por conta da dízima
  então $124,45_{(10)} = 147,404040\ldots_9$

- normalizar e achar o expoente
  $147,4040_9 = 147,4040_9 \times 9^0$
  $147,4040_9 = 14,74040_9 \times 9^1$
  $147,4040_9 = 1,474040_9 \times 9^2$
  $147,4040_9 =, 1474040_9 \times 9^3$
  o expoente é 3 e está dentro do limite.
- tamanho definido pela máquina ($t = 3$) e o número fica $0,147$ ou 147
- erro absoluto: reconverter 147_9 para seu valor decimal $1 \times 9^2 + 4 \times 9^1 + 7 \times 9^0$
  $81 + 36 + 7 = 124,0$
- erro absoluto: $|124,45 - 124| = 0.45$
- erro relativo: $\frac{0.45}{124} = 0.00362 \approx 0.0036$

Python

```
# maquina genérica
import numpy as np
def maq(beta,t,a,b,numero):
  pi=np.floor(numero)
  pf=numero-pi
  nb=[]
  while 1==1:
    nb=[pi%beta]+nb
    pi=pi//beta
    if pi==0:
      break
  fb=[]
  ct=0
  while ct<9:
    pf=pf*beta
    hora=np.floor(pf)
    fb=fb+[hora]
    pf=pf-hora
    ct=ct+1
  nu=nb+fb
  nu=nu[0:t]
  tamanho=len(nb)
  expoente=tamanho
  if expoente<a:
    print('underflow')
    return
  if expoente>b:
    print('overflow')
    return
  print('numero = 0.',end='')
  for i in range(len(nu)):
    print(int(nu[i]),end='')
  print(' * ',beta,' ^ ',expoente)
  bacana=0
  for i in range(len(nu)):
    bacana = bacana +
      (nu[i]*(beta**(tamanho-1)))
    tamanho=tamanho-1
  ea=np.around(np.abs(bacana-numero),4)
  er=np.around((ea/bacana),4)
  print('erro absoluto = ',ea)
  print('erro relativo = ',er)
  return

maq(6,5,-4,4,124.45)
```

com a resposta

```
numero = 0.32424 *  6  ^  3
erro absoluto =  0.0056
erro relativo =  0.0
```

☞ Para você fazer

A seguir, 4 instâncias do mesmo problema (armazenar um número e depois calcular o erro absoluto neste armazenamento.

	β	t	$[a$	$b]$	número a converter
1.	7	4	-9	9	4364.35
2.	5	5	-9	9	15988.8
3.	6	3	-9	9	2246.53
4.	8	3	-9	9	739.295

Responda aqui

	mantissa	erro absoluto
1		
2		
3		
4		

```
===== 11/06/2022 13:26:59.6 =====E=PLk08
    1       343  2.9000   57625   .0150   10763  7.2000    1213   .5500
    2       201  1.3500   32302   .1327    2212   .8000     112 21.9270
    3     11561   .0800    1101111.7000  107513   .1050     112 11.2400
    4    201160   .0069   68573   .0012  321204   .0200     674   .4300
    5      1550  3.3500   10024 13.8000     142 14.5300     134  3.2950
```