Dízima Periódica

Praciano-Pereira, Tarcisio [1]

30 de setembro de 2023
preprints da Sobral Matemática
no. 2023.03
Editor Tarcisio Praciano-Pereira
tarcisio@sobralmatematica.org

[1]tarcisio@sobralmatematica.org

Resumo

As *dízimas periódicas* são os números racionais e as *dízimas não periódicas* são os números irracionais. Eu vou mostrar como se produzem as geratrizes das *dízimas periódicas*, usando exemplos e ao final vou apresentar um programa que recebe os dados duma dízima e cria a sua geratriz.

palavras chave: dízima periódica, geratriz, número racional.

I will discuss periodic decimals, or repeating decimal, as numbers that have a fraction which generate them and I will show how this fraction can be constructed. At the end I will give out a program to create the generating fraction of a repeating decimal.

keywords: periodic decimals,rational numbers repeating decimal.

0.1 Objetivo

Neste artigo estou tratando das dízimas periódicas que são os números racionais e tem uma geratriz, uma fração. Primeiro vou dar exemplos e construir as geratrizes destes exemplos. Depois vou fazer uma classificação das dízimas periódicas caminhando para a construção do algoritmo.

As dízimas periódicas são séries geométricas com razão menor do que 1 portanto são séries convergentes o que vai me levar a falar de limite. Vou então relembrar a soma das progressões geométricas para finalizar com a fórmula da soma das séries geométricas.

Ao final vou mostrar dois programas, um escrito em `calc`[Lo11] e outro escrito em `python3` que reproduzem a dízima usada como entrada de dados e apresenta a geratriz da mesma.

Mostrei este artigo a um amigo que me disse rindo que eu poderia obter um programa muito eficiente se o solicitasse um código a um desses plagiários que se passam por IA e eu teria algo melhor do que aquele que eu houvesse escrito. E fizemos isto juntos usando o android dele, obtivemos um programa em `C++` e outro em `python` bem parecidos com o meu, nos dois casos o programa pedia `parte não periódica`, `parte periódica` e localização do ponto decimal, exatamente como os meus programas fazem.

Então eu tentei melhorar os meus programas me apoiando nas respostas do *plagiário* e rapidamente me dei contas de que é um problema de classe NP, provavelmente. Dada uma dízima periódica qualquer, tanto a *parte periódica* como *parte não periódica* podem ter qualquer tamanho, então, se conseguisse escrever o programa e lhe entregasse algumas páginas contendo estas duas respostas, não codificadas, mas sob a forma duma dízima periódica, o meu computador levaria anos trabalhando para conseguir separar as duas partes, mesmo duma dízima cujo período tivesse alguns milhares de dígitos e ainda seria um pequeno caso particular ...

Eu desisti de melhorar o meu programa, e deixo-o como se encontra ao final porque ele é semelhante ao oferecido pelo *plagiário* que se passa por *inteligência artificial*, qualquer que seja ele. Eu com minha *inteligência natural* obtive o mesmo resultado.

0.2 Exemplos para uma classificação

Dizima periódica, é um caso particular de *dízima* que é uma sucessão de algarismos definindo um número. Dízima é sinônimo de *decimal* embora pouco usada com este significado. Neste artigo vou tratar das *dízimas periódicas* que são os números racionais.

As *dízimas periódicas* são os números racionais e as *dízimas não periódicas* são os números irracionais, vou referir-me a estas últimas simplesmente como *dízimas* e elas estão fora do meu objetivo neste artigo.

Alguns exemplos sobre os quais farei comentários em seguida são

$$2 = 1.(9)\cdots \in \mathbf{Q} \text{ dizima periódica;} \tag{1}$$
$$1/3 = 0.(3)\ldots; \in \mathbf{Q} \text{ dizima periódica;} \tag{2}$$
$$0.(1415)\cdots \in \mathbf{Q} \text{ dizima periódica;} \tag{3}$$
$$4567123.(1415)\cdots \in \mathbf{Q} \text{ dizima periódica;} \tag{4}$$
$$4567.123(1415)\cdots \in \mathbf{Q} \text{ dizima periódica;} \tag{5}$$
$$p = 3.(14159265358979323848)\cdots \in \mathbf{Q}; p \neq \pi; \tag{6}$$
$$\pi \approx 3.14159265358979323848\ldots \notin \mathbf{Q}; \tag{7}$$

Nas equações (eq.1), (eq.6), você vê dízimas periódicas. na equação (eq.7) você tem a *constante de Arquimedes*,π, que é um *número irracional* não é uma dízima periódica.

Na equação (eq.6), eu criei uma *dízima periódica*, p, que é uma aproximação da *constante de Arquimedes*,π, mas observe que ao final da equação eu deixei a observação $p \in \mathbf{Q}; p \neq \pi$, p não é π. Mais adiante eu vou construir uma *geratriz* para esta dízima, uma aproximação do número π. Mas deixe-me trabalhar com o padrão que se encontra nas equações (eq.3) até (eq.5) porque destes três casos eu vou obter facilmente o algoritmo bem conhecido para obter a geratriz duma dízima qualquer.

O caso da equação (eq.3) é o padrão que vou usar para obter os outros dois.

As dízimas se classificam em três classes que você pode ver imediatamente na próxima lista de equações, entretanto o tipo básico, que aparece em todos os outros casos está na equação (eq.3) e vou começar a discussão por este caso mais simples produzindo ao final uma sua generalização para descrever todos os outros. Eu explico porque vou escolher este caso mais simples, Deixe-me estabelecer uma *linguagem*, uma *dízima periódica* é formada

- do período, a parte que se repete,
- mais um número, a parte não periódica.

O caso mais simples, (eq. 3), quando a *parte não periódica* for zero e sucessão de períodos começa logo depois do ponto decimal. Basta multiplicar ou dividir por uma potência de 10 para cair neste caso. Confira os exemplos

$$p = 4567.123(1415)\cdots \mapsto 1000p = 4567123.(1415); \tag{8}$$
$$p = 4567123.(1415)\cdots \mapsto p - 4567123 = 0.(1415); \tag{9}$$
$$p = 0.4567123(1415)\cdots \mapsto 10000000p = 4567123.(1415); \tag{10}$$
$$p = 0.4567123(1415)\cdots \mapsto 10000000p - 4567123 = 0.(1415); \tag{11}$$

Deixe-me prosseguir com a equação (eq.9) que é o mesmo obtido na transformação da equação (eq.11) e nestes exemplos estou mostrando como passei para o caso mais simples fazendo uma transformação que é inversível. No caso da equação (eq.9) basta-me somar 4567123 à geratriz encontrada para ter a geratriz da dízima original. No caso da equação (eq.11) tenho que somar 4567123 e depois dividir por 10000000 à geratriz encontrada para ter a geratriz da dízima original.

0.3 Construindo o algoritmo

Em $p = 0.(1415)\cdots$ o período tem 4 dígitos portanto é uma sucessão de números da forma $\frac{1415}{10^4} + \frac{1415}{10^8} + \ldots$, Observe o que acontece quando eu uso uma calculadora:

$$1415/10000 = 0.1415; \tag{12}$$
$$1415/9999 = 0.14151415141514151415\cdots = 0.(1415); \tag{13}$$
$$3/10 = 0.3; \tag{14}$$
$$3/9 = 0.33333333333333333333; \tag{15}$$

Nos dois casos eu substitui o denominador por um número menor, uma sucessão de "noves", consequentemente obtive uma fração maior, porque uma sequência de n noves é um número menor do que 1 seguido de n zeros.

Eis o método para obter a geratriz quando a parte periódica vem logo depois do ponto decimal: *uma fração tendo por numerador um período e por denominador tantos noves quantos forem os algarismos do período.*

Você pode ver a razão de porque no algoritmo aparece um número "contendo tantos noves quantos sejam os dígitos do período", no denominador porque $9999 \approx 10000$ e na última equação $9 \approx 10$. Agora *eu preciso provar que é assim* porque alguns exemplos nada provam, eles apenas ajudam a compreender a demonstração que vou produzir.

Afinal, o que é uma *dízima periódica*? Como todo número escrito em forma decimal, é uma combinação linear de potências de 10.

$$12344 = 10^4 + 2*10^3 + 3*10^2 + 4*10 + 4*10^0; \tag{16}$$
$$123.44 = 10^2 + 2*10^1 + 3*10^0 + 4*10^{-1} + 4*10^{-2}; \tag{17}$$
$$123.44 = 10^2 + 2*10 + 3 + 4*10^{-1} + 4*10^{-2}; \tag{18}$$

As duas últimas equações trazem o mesmo exemplo, no segundo caso na forma usual, ninguém escreve $3*10^0$ e sim 3.

Do exemplo você observa que o *ponto decimal*, ou a *vírgula*, no modo de escrever brasileiro, vem antes da primeira potência negativa, quando um número for menor do que 1 será uma soma de potências negativas de 10. No caso das *dízimas periódicas* eu tenho um bloco de mesmo tamanho que se repete, nos exemplos acima eu peguei um pedaço da expressão do π para construir *dízimas periódicas*, 1415, blocos com quatro algarismos. Daria exatamente no mesmo com uma quantidade n qualquer de algarismos, apenas me daria mais trabalho para escrever e π é um caso interessante...

A expressão decimal dum número é sua expansão em potências de 10, a expressão binária é sua expansão em potências de 2, e ainda outra forma mais usada é *hexadecimal* em que se usa a expansão em potências de 16. Há outras forma de expressar os números que não interessam para o objetivo deste artigo em que vou me concentrar no caso decimal.

Como uma *dízima periódica* é uma soma de potências de 10 então o modelo para elas é a *progressão geométrica* e vou começar relembrando a soma dos termos duma *progressão geométrica* de razão $0 < r < 1$, a razão é um número positivo menor do que 1 que é caso das potências negativas de 10.

Então

$$1 + r + r^2 + \cdots + r^n = \frac{r^{n+1}-1}{r-1}; r + r^2 + \cdots + r^{n+1} = r\frac{r^{n+1}-1}{r-1}; \tag{19}$$
$$p = 0.(1415) \approx 0.(1415)(1415)(1415)(1415)\ldots; \tag{20}$$
$$p \approx \frac{1415}{10^4} + \frac{1415}{10^8} + \cdots = 1415\sum_{k=1}^{k=n+1} r^k = 1415r\sum_{k=0}^{k=n} r^k = 1415r\frac{r^{n+1}-1}{r-1}; r = \frac{1}{10^4}; \tag{21}$$
$$1415r\frac{r^{n+1}-1}{r-1} \mapsto 1415r\sum_{k=0}^{\infty} r^k = 1415r\frac{-1}{r-1} = \frac{1415r}{1-r}; \tag{22}$$

a soma dos termos duma p.g. de razão $r = \frac{1}{10^4}$. Mas a "*soma finita*" é um valor aproximadopara p e tenho aplicar *limite* quando n *for infinito* para obter o valor exato de p. É aqui um dos momentos em que é possível começar o estudo do *limite* no Cálculo, eu acho que é o melhor momento, mas isto depende do *método pedagógico de cada docente.*

A linguagem comum é dizer n tende para infinito "como" se n se movesse. Limite é o valor no infinito...

Aplicando *limite* na segunda parte da equação (eq.21) eu tenho

$$p = 1415 \times r\frac{-1}{r-1} = \frac{1415\times r}{1-r} = \frac{1415r}{1-r}; \tag{23}$$
$$1 - r = 1 - \frac{1}{10^4} = \frac{10^4-1}{10^4} = \frac{9999}{10000}; \frac{1}{1-r} = \frac{10000}{9999}; \tag{24}$$
$$p = \frac{1415\times 10000}{99990000} == \frac{14150000}{99990000} = \frac{1415}{9999}; \tag{25}$$
$$0.333\cdots = \frac{3}{9}; \tag{26}$$

Além disto eu cometi um erro no parágrafo precedente. Eu deixei o erro marcado com aspas: somente existem somas *finitas* então o adjetivo é desnecessário e confuso. Falei de *limite* que é o salto lógico que nos libera das amarras da Aritmética, então, não existem *somas infinitas* e sim o *limite* duma sucessão de *somas*.

A geratriz de uma dízima constituída apenas do período depois da vírgula, é uma fração tendo por numerador um período e por denominador tantos noves quantos sejam os algarismos do período. A quantidade de noves vem da quantidade de zeros de 10^n porque n é quantidade de algarismos dum período: $10^n - 1 = 999\ldots9$ tem n "*noves*". Apenas com estes passos, e fazendo as modificações sugeridas nas equações (eq.12) (eq.15) você já pode calcular a geratriz de qualquer dízima periódica e nem precisa ler o resto do artigo. Mas, aguarde, *eu vou construir o algoritmo ao final*!

0.4 Geratriz que a constante de Arquimedes

Logo vou mostrar-lhe, antes de continuar com o projeto, como eu posso construir uma fração, um número racional, uma dízima periódica, que me dê um *valor muito aproximado para* $\pi \notin \mathbf{Q}$. Usando uma linguagem de programação peço o calculo $atan(1)$, o ângulo cuja tangente seja 1, que é $\frac{\pi}{4}$, então peço que a linguagem de programação escreva $4 * atan(1)$, e tenho

$$4 * atan(1) -> 3.14159265358979323848; \tag{27}$$

as linguagens de programação, ou as calculadoras, têm um limite de precisão, eu usei `calc`[Lo11] que tem uma precisão de 20 casas decimais.

Suponha que esta expressão seja 3 seguido de um período, neste caso o "período" tem 20 algarismos,

$$4 * atan(1) -> 3.(14159265358979323848) \tag{28}$$

A constante de Arquimedes,π, não é um número racional e eu estou usando uma aproximação produzida por uma calculadora que vou supor que seja uma `dízima periódica`. Então a geratriz seria

$$3 + \frac{14159265358979323848}{99999999999999999999}; \tag{29}$$

$$3 * 99999999999999999999 + 14159265358979323848 = 314159265358979323845; \tag{30}$$

$$\frac{314159265358979323845}{99999999999999999999}; \tag{31}$$

E se eu colocar esta expressão de volta na linguagem de programação eu vou obter

$$3.14159265358979323848 \tag{32}$$

As linguagens de programação não saberiam escrever

$$3.(14159265358979323848) \tag{33}$$

o número 3 seguido de um período mas eu sei fazer as contas, *com minha inteligência humana*, e fiz.

Na equação (eq. 29) você tem a geratriz de *uma dízima periódica* que lhe fornece uma aproximação para π com um período de vinte casas decimais. Raspe e cole a fração que aparece na última equação, num terminal do `python`, ou de sua *linguagem de programação interpretada favorita*, e você verá como resultado

$$3.14159265358979323848\ldots \tag{34}$$

Este último exemplo do cálculo duma dízima periódica que forneça uma aproximação para π lhe mostra como funciona o algoritmo quando eu tiver número inteiro seguido de um período logo depois do *ponto decimal*.

Você já tem aqui todos os passos necessários para seguir calculando geratrizes de dízimas periódicas e *produzir o seu próprio algoritmo* sem precisar de continuar a leitura deste artigo. Mas, por favor, aguarde que vou construir o algoritmo ao final!

0.5 Quando houver parte não periódica

Se houver uma parte não periódica eu vou supor que ela fique antes do ponto decimal, antes da *a vírgula*. No meu método de construção da Matemática, eu vou sempre construindo a partir de exemplos mais simples de onde vou tirando, aos poucos, a regra principal até obter o caso geral.

Vou aplicar o método que usei para obter a equação (eq.34) e depois somo a parte inteira à geratriz obtida e vou ajustar o resultado multiplicando ou dividindo por 10^n que vai *correr* o *ponto decimal* para local exato. O algoritmo vai traduzir isto mencionando a presença de zeros

- no *numerador* ou no *denominador*,
- que é como correr o *ponto decimal*
- para *direita*, um número maior, ou para *esquerda*, um número menor.

Exemplo 1 (Quando a parte) *periódica começa depois da vírgula*

- *Caso a parte periódica comece com alguns algarismos que fiquem à esquerda do ponto decimal,* $31415.(1414)$*, primeiro eu divido por* 10^n*, com tantos zeros quantos sejam os algariamos da parte periódica que se encontrem à esquerda, divido por* 10^n*, para que todos os períodos fiquem depois do ponto decimal,* $3.(1414)$*. Então eu me encontro no caso mais simples, calculo a geratriz, Depois multiplico a geratriz pelo mesmo valor,* 10^n*.*
- *Caso haja parte não periódica fique à direita do ponto decimal,* $0.123456789(1414)$ *primeiro multiplico por 10 seguidos de tantos zeros quantos sejam os algarismos da parte não periódica que fica depois do ponto decimal,*10^n*, o que me leva a um dos casos anteriores,* $123456789.(1414)$ *depois divido o resultado por* 10^n*.*

Existe um algoritmo simples para recuperar a forma $\frac{p}{q}$ que representa a mesma dízima. Vou mostrar-lhe um exemplo a partir do qual vou deduzir o algoritmo que vai ser ligeiramente diferente daquele que aparece comumente nos livros de Matemática do Ensino Médio. Este que estou apresentando aqui é bem intuitivo e sua demonstração obedece às regras da lógica expandido as regras da Aritmética, eu vou fazer a apresentação fazendo alterações no exemplo.

A justificativa do algoritmo vem das *séries geométricas* que generalizam as *progressões geométricas* porque uma dízima periódica representa uma soma de potências negativas de 10, com tantos zeros quantos algarismos houver na *parte periódica*, somada com a parte não periódica. Usando a linguagem das p.g. a *parte não periódica* é o *termo inicial* da p.g. a *parte periódica* é formada por um *coeficiente constante*, o período, que multiplica a razão que é uma potência negativa de 10. Resumindo esta descrição, uma *dízima periódica* é um *termo inicial* somado com uma *série geométrica* cuja razão é uma potência negativa de 10 contendo tantos zeros quantos sejam os dígitos do período.

0.5.1 Voltando ao caso do π

Vou construir a "geratriz" para a dízima $3.(1414)$, uma aproximação da *constante de Arquimedes* com um período de 4 algarismos. Vou apenas listar os passos deixando na última equação a expressão que você pode raspar e colar num terminal duma linguagem de programação, como `python`.

$$p = 3.(1415); 3 * 9999 = 29997; \tag{35}$$
$$\frac{1415}{9999} + 3 = \frac{1415}{9999} + \frac{29997}{9999}; \tag{36}$$
$$29997 + 1415 = 31412; \tag{37}$$
$$p = \frac{31412}{9999}; \tag{38}$$

Raspe e cole a última expressão num terminal de sua *linguagem de programação interpretada favorita* e você terá uma dízima periódica com período de 4 algarismos que é uma aproximação de π.

0.5.2 A notação das dízimas periódicas

Não há uma notação padrão para indicar a parte periódica. Alguns autores usam parenteses, como eu venho usando. Outros marcam a parte periódica com um barra acima da mesma e alguns usam uma barra inferior.

0.5.3 Usando limite

Da equação (eq. 19) até (eq. 21) relembrei a *teoria* das progressões geométricas com uma adaptação da linguagem para o meu objetivo com as dízimas periódicas.

Na equação (eq. 22) eu escrevi o limite da *série geométrica* que é *o limite das somas* dos termos da progressão geométrica que sendo uma p.g. com razão positiva menor do 1, tem limite, a série converge para um número racional $\frac{p}{q}$ a *geratriz da dízima periódica*.

Observe que eu interrompi uma cadeia de igualdades substituindo uma delas pelo símbolo $\mapsto$ marcando o *salto lógico* representado pela aplicação do *limite* que não é uma operação da Aritmética o que estou caracterizando como um *salto lógico*: há uma sequência de operações da *Aritmética* seguidas duma operação que não é *Aritmética*, o *limite*.

O *somatório* com o símbolo "∞" não é uma soma, é um *limite*, é apenas um símbolo que representa o *limite* que eu escrevi no final da equação (eq.22) e que vou usar nas equações seguintes para obter a fórmula da *geratriz* neste caso particular e que vai me permitir generalizar para um caso qualquer quando vou obter o algoritmo para obter a *geratriz* duma dízima periódica qualquer. É neste ponto que, *com um salto lógico*, usando limite, estou expandindo as regras da *Aritmética* porque este somatório não representa uma *soma da Aritmética* e estou usando *limite*. É possível mostrar que as *operações da Aritmética* valem para o *limite*. Isto será explorado na construção dos números reais que é *o conjunto dos limites e onde também valem as operações da Aritmética*. É com referência ao conjunto $\mathbf{R}$ que estou me referindo ao *salto lógico*. Há um *salto lógico* na passagem de $\mathbf{Q}$ para $\mathbf{R}$. $\mathbf{Q}$ está no domínio da *Aritmética* e $\mathbf{R}$ está fora do domínio da *Aritmética* mas valem todas as regras da *Aritmética* para os números reais.

Limite é um salto lógico!

0.6 algoritmo para recuperar a geratriz

Vou usar *algoritmo* no caso particular exposto nas equações acima que é bem simples porque o *ponto decimal*, a *vírgula*, já separa, identificando as duas partes da dízima, a *não periódica*, antes do ponto e a *periódica*, depois do *ponto decimal*.

1. Nas equações iniciais você viu que pode haver um *valor inicial*, *a parte não periódica*, que logo ignorei caindo num caso mais simples, mas agora vou voltar a considerá-lo:

$$S = 123456789.14151415\cdots = 123456789.(1415)\ldots \text{ valor inicial: } 123456789; \quad (39)$$

2. Identifique as partes: periódica e não periódica: 123456789, 1415, são duas *listas* de números que tem, respectivamente, os comprimentos $9, 4$, $|123456789| = 9, |1415| = 4$, neste exemplo!

3. Crie uma fração, em cujo *numerador* esteja a *parte periódica* e no denominador uma *lista* de 9s com o comprimento da lista que define o período: 9999, com tantos 9s quantos forem os algarismos da *parte periódica*. Se você digitar esta fração numa folha de trabalho do `calc` ou do `python` vai aparecer a parte decimal da *dízima*:

   - Em `python2` "1415/9999." e se você digitar "1415/9999" vai ter uma surpresa, experimente e procure entender!
   - Em `calc`[Lo11], é uma `linguagem de programação` que gosto de usar,
     1415/9999
     ~ 0.14151415141514151415141514151415141514151415
     O sinal "~ " é a forma como `calc` o adverte de que se trata duma aproximação porque *ele não sabe escrever*
     0.14151415141514151415141514151415141514151415…

A divisão no `python2` é inteira. Isto foi modificado em `python3`, portanto depende de que versão do `python` você esteja usando. Em `python2` digite usando o `ponto decimal` no final da lista de "noves".

4. Agora deixe-me terminar as contas na equação (eq.39).

$$123456789 + \frac{1415}{9999} = \frac{123456789 * 9999 + 1415}{9999} = \frac{373412655 + 1415}{9999} = \frac{373414070}{9999}; \quad (40)$$

e pode testar com `python` ou `calc`, [Lo11].

0.6.1 `calc` e `python3`

Se você usar `Linux`, e se a distribuição for baseada no `Debian`, `Debian/gnu/linux`, você pode instalar `calc` com o comando

`sudo apt install apcalc`.

sem precisar pagar nada aos autores do programa porque `calc` é um programa livre distribuído com a licença `GPL`. Esta afirmação nada tem o que ver com *gratuidade* por que isto simplesmente não existe. Nós que escrevemos programas e os distribuímos sob `GPL`, trabalhamos para alguma empresa que nos permite produzir programas sob esta licença, no meu caso eu sou professor universitário e posso fazê-lo. Outros programadores pedem que lhes seja feita uma doação que lhes permitam continuar trabalhando de acordo com `GPL`, fora do horário de trabalho da empresa. Não sei qual é o caso do `calc`, mas digitando `help copyright`, num terminal do `calc` você vai obter alguma informação que lhe pode ajudar descobrir mais a respeito. Você pode instalar `python3` com o comando

Sou professor aposentado!

`sudo apt install python3`

mas provavelmente já estará instalado porque `python3` é considerado uma linguagem nata nas distribuições `Debian/gnu/linux` e neste caso você vai receber a mensagem de que `python3` já é a versão mais recente instalada no sistema. Dentro do terminal do `python3` digite `copyright()` e você pode ler as informações sobre seus direitos de uso da linguagem. Num terminal do `calc` digite `help copyright` e você pode ler sobre seus direitos de uso da linguagem.

0.7 Quando o ponto decimal não identifica as partes

Aqui você tem três casos que são diferentes do exemplo inicial. Vou usar dois parâmetros para discutir os exemplos,

- m para a quantidade de zeros, que vai regular a posição do *ponto decimal*, da vírgula, e
- n para a quantidade de noves,

Eu já vinha usando o parâmetro n e agora vou usar o parâmetro m que vai me ajudar a correr o ponto decimal para o local correto.

n é a potência de 10 em $r = \frac{1}{10^n}$ a razão da *série geométrica*, e a potência de 10 indica a quantidade de noves que vai aparecer no denominador da dízima.

Uma *progressão geométrica* é uma sucessão finita para a qual vale falar na soma dos termos da *progressão geométrica*. Uma *série geométrica* é uma sucessão infinita, já não posso mais falar em "soma dos termos duma *série geométrica*", porque para a Aritmética não existem "somas infinitas". Estou tratando do *limite* da sucessão de soma dos termos duma *progressão geométrica* que existe quando $r < 1$ e que é o caso das *dízimas periódicas*.

Então a *geratriz* duma dízima periódica é um *limite*.

Não se confunda! Estou sempre usando o mesmo exemplo com o período (1415)!

Exemplo 2 (Ponto decimal) *deslocamento da vírgula*

1. $123456789.(1415)\ldots\ m = 0; n = 4;$

2. $1234567891415.(1415)\ldots\ m = 4; n = 4;$ *ou*

3. *poderia ser* $0.01234567891415(1415)\ldots\ m = -10; n = 4;$

E você vai poder obter a *dízima periódica* com o comando do `calc`

```
power(10,m)*(123456789*9999 + 1415)/9999
```

usando o valor de m apropriado a cada caso. Por exemplo, para o terceiro caso usei

```
m=-10; power(10,m)*(123456789*9999 + 1415)/9999
```

Nesta linha anterior se encontram dois comandos do `calc`, separados por "ponto e vírgula", ou poderia serem os comandos separados em linhas distintas. Se você raspar a linha de comandos acima, e a colar num terminal do `calc` vai ter na frente dos olhos a dízima periódica do terceiro exemplo. Troque o valor de m e você pode obter os outros dois. Com `python3` troque `power` por `pow` que é o comando para calcular potências em `python3`. A precisão do `python3` é ligeiramente pior do que a precisão do `calc` e você deve notar esta diferença que de certa forma explica porque eu prefiro `calc` mas muita gente prefere `python3`.

Eu obtive as *dízimas* e agora tenho que traduzir as operações para obter as *geratrizes*, como eu já fiz acima para obter uma *geratriz* aproximada para π, usando a minha *inteligência humana*. Também usando a *minha inteligência não artificial* eu posso construir um programa para repetir as passos que vou agora identificar para construir qualquer *geratriz* usando uma linguagem de computador, quando então eu estarei entrando num processo colaborativo com milhares de outros programadores para construir um algoritmo, "*subindo nos ombros de gigantes*".

Vou usar o exemplo inicial da *dízima periódica* para construir sua *geratriz* e desta forma generalizar esta construção para uma dízima qualquer e assim produzir um algoritmo.

1. Deixe-me caracterizar o primeiro dos exemplos acima como o *padrão*, quando $m = 0$, e os dois outros como modificações do *padrão* em que o *ponto decimal* corre para direita ou para esquerda como conveniente for.

2. Identifique a *parte periódica* e divida-a por uma lista de 9s com mesmo comprimento do período.

3. Some o resultado anterior com a *parte não periódica* multiplicada pela lista de 9s com mesmo comprimento do período. Observe que você estará somando frações e o processo é aquele para igualar os denominadores,
$$A + \frac{B}{9999} = \frac{A9999 + B}{C}; \tag{41}$$
a lista de noves na equação anterior representa um denominador comum na soma duma fração com o número A que representa a parte não periódica e B representa o período. Divida agora este resultado pela lista de 9s que é o *denominador comum*.

4. Multiplique por uma potência de 10 (o expoente pode ser negativo). É esta multiplicação que vai governar a colocação do *ponto decimal*. Na prática, desloque o ponto decimal para voltar ao caso anterior e conte o número de casas correspondentes a este deslocamento do *ponto*, como foi feito no exemplo 2, na página 7 e seja m este número que *governa o deslocamento do ponto decimal* e vai ser
   (a) $m = 0$ no primeiro caso, não há deslocamento do ponto decimal,
   (b) $m = 4$ no segundo caso, o ponto decimal se deslocou quatro casas para a direita,
   (c) $m = -10$, no terceiro caso, o ponto decimal se deslocou dez casas para a esquerda, deixando um zero depois do *ponto decimal*.

Aqui esta o uso dos dois parâmetros, que vou precisar para construir o programa, um algoritmo, como ficou indicado dentro da lista de casos do exemplo 2, na página 7

Uma outra forma prática, relativamente ao deslocamento do *ponto decimal*, ou a vírgula, consiste em colocar m zeros no denominador ou no numerador. É o papel de `power(10,m)` no algoritmo computacional.

A geratriz vai ser a soma de frações usando o primeiro caso como padrão e usando n para retornar à *dízima periódica* original. Vou repetir a lista de casos com o algoritmo computacional para produzir a dizima, ainda não a geratriz!

1. $123456789.(1415)\ldots\ m = 0$;

   `power(10,m)*(123456789*9999+1415)/9999;`

   o ponto decimal não se deslocou nem para direita e nem para esquerda.

2. $1234567891415.(1415)\ldots\ m = 4$;

   `power(10,m)*(123456789*9999+1415/9999);` o ponto decimal se deslocou para esquerda de quatro casas decimais.

3. poderia ser $0.01234567891415(1415)\ldots\ m = -10$ o ponto decimal se deslocou para direita de 10 casas decimais, deixando um zero depois do ponto decimal.

   `power(10,m)*(123456789*9999+1415)/9999;`

Neste ponto *eu cometi um erro* que vou registrar como uma curiosidade para quem não tiver tanta experiência com uso de expressões computacionais, (e eu tenho experiência, e errei):

Havia um erro! Corrigido!

1. errado `power(10,m)*(123456789*9999+1415/9999)`,

2. certo `power(10,m)*(123456789*9999+1415)/9999`,

o erro se encontra na posição do parêntesis à esquerda cortando a distributividade na *multiplicação* por $\frac{1}{9999}$.

Nas listagens dos três exemplos você vê comandos para a linguagem de computação `calc`, ou `python`, e eu preciso agora lhe mostrar onde esta a *geratriz* e como obtê-la. Correr o ponto decimal para a direita ou para esquerda, significa colocar zeros no denominador ou no numerador, nesta ordem. É o papel do multiplicador `power(10,m)`. Então eu vou repetir a lista de casos agora construindo a *geratriz*:

1. $123456789.(1415)\ldots\ n = 0$;

$$\frac{123456789 * 9999 + 1415}{9999} = \frac{373414070}{9999} \tag{42}$$

   e o comando do `calc`

   `power(10,m)*(123456789*9999+1415)/9999;`

2. $1234567891415.(1415)\ldots\ n = 4$;

$$10^4 \frac{123456789 * 9999 + 1415}{9999} = 10^4 \frac{373414070}{9999} = \frac{3734140700000}{9999} \tag{43}$$

   Os zeros são incluídos no final do numerador na expressão da geratriz e o comando do `calc`

   `power(10,m)*(123456789*9999+1415)/9999;`

3. poderia ser $0.0123456789(1415)\ldots$ $n = -10$
$$10^{-10}\frac{123456789 * 9999 + 1415}{9999} = 10^{-6}\frac{1234444434625}{9999} = \frac{37341407}{9999000000} \quad (44)$$
e o comando do `calc`
`power(10,-10)*(123456789*9999+1415)/9999;`

Use o algoritmo anterior e multiplique o resultado por 10^m em que $m = -10$ é negativo, quando o deslocamento do *ponto* foi para direita, e $m = 4$, positivo, no outro caso, quando o ponto foi deslocado para a esquerda.

Experimente usando `calc`, para obter a geratriz de $1234567891415.(1415)\ldots$
`m=-4; power(10,m)*(123456789*9999 + 1415)/9999`
A geratriz é o resultado da conta
`power(10,m)*(123456789*9999 + 1415)/9999` que é $\frac{1234444434626}{99990000}$ e você também pode verificar usando `calc` ou outra linguagem de processamento interpretada.

Separe os *processos*:

- para obter a geratriz, calcule com `calc` o numerador: `123456789*9999 + 1415` e escreva à mão a fração $\frac{1234444434626}{9999} * 10^m$,
- para recuperar a dízima periódica use o comando duplo do `calc`
  `m=4; power(10,m)*(123456789*9999 + 1415)/9999` e a geratriz é $\frac{12344444346260000}{9999}$ substituindo "m=4" com o valor apropriado.
  No terceiro caso a fração é $\frac{1234444434626}{99990000000000}$ correspondente à dízima $0.0123456789(1415)$.

E você pode ver que (o algoritmo) *começa se configurar*,

1. escreva uma fração cujo denominador seja uma *lista* de noves do mesmo comprimento do período.
2. o numerador seja a parte não periódica multiplicada pela referida *lista* de noves somado a um período.
3. acrescente m zeros ao numerador se o ponto decimal estiver deslocado m posições para esquerda, ou
4. acrescente m zeros ao denominador se o ponto decimal estiver deslocado m posições para direita.
5. O acréscimo de zero equivale a multiplicar por 10^m que vai correr o ponto decimal, a vírgula, para o local apropriado.

Experimente usando `calc`

`power(10,-5)*(123456789*9999 + 1415)/9999;` (45)

123456789 + 1415/9999 = (123456789*9999 + 1415)/9999; (46)

(123456789*9999 + 1415)/9999 = 1234444434626/9999; (47)

`power(10,-5)*1234444434626/9999;` (48)

1234.56789(1415)... (49)

$\frac{1234444434626}{999900000} = 1234.56789(1415)\ldots$ (50)

Na folha do `calc`, digite: (51)

1234444434626/999900000; (52)

Compare o valor de m com o número de zeros que aparece no denominador da geratriz.

A geratriz é o resultado da conta

`123456789 + 1415/9999` que é $\frac{1234444434626}{999900000}$ onde você pode identificar a multiplicação por `power(10,-5)` na presença de cinco zeros depois dos quatro *nove*s, no denominador.

`power()` é a função do `calc` para calcular qualquer potência, inclusive fracionária. Usando `python3`, substitua `power()` por `pow()`.

Em alguns textos aparece a fórmula errada para determinar a geratriz duma dízima periódica como sendo:

1. no numerador *uma lista formada com a parte não periódica seguida dum período*
2. no denominador *uma lista de 9 com o mesmo comprimento da parte periódica seguida por uma lista de 0 com o mesmo comprimento da parte não periódica*

e está errado, como o mostram os exemplos anteriores.

0.7.1 A fração obtida pode ser redutível

Este método produz a geratriz de qualquer dízima periódica, entretanto o resultado pode não ser a fração mais simples. Por exemplo

$$0.(142857)\cdots = \frac{142857}{999999} = \frac{1}{7}; \tag{53}$$

E você pode gerar vários exemplos interessantes, tome um número primo, p, multiplique-o por uma lista de noves somando 1 ao resultado para obter o numerador, coloque a lista de noves no denominador e vai obter uma *geratriz* da dízima periódica. Por exemplo

$$\frac{p * 99999 + 1}{99999} \tag{54}$$

em que p é primo.

O comando do `calc` é

`p=5; (p*99999+1)/99999`

para obter

$$5.00001000010000100001\cdots = 5(00001)$$

Como o denominador é sempre divisível por 3 basta verificar se o numerador também o é para obter uma fração mais simples. No caso da equação (eq. 54), eu fatorei o numerador

$$142857 = 3^3*11*13*37; 999999 = 3*333333 = 9*111111 = 3^3*7*11*13*37 = 7*142857; \tag{55}$$

e eliminei os fatores comuns com o denominador.

As duas frações

$$\frac{142857}{999999} \approx \frac{1}{7} \tag{56}$$

são *equivalentes* portanto qualquer uma das duas é uma boa resposta para a questão da procura duma geratriz para uma dízima periódica.E por que? A resposta está aqui

$$999999 = 3*3*3*7*11*13*37; \tag{57}$$
$$142857 = 3*3*3*11*13*37; \tag{58}$$

simplificando fica $\frac{1}{7}$ e é difícil escrever um algoritmo que faça esta simplificação, qualquer algoritmo vai terminar escrevendo $\frac{142857}{999999}$ deixando a simplificação por sua conta . . .

0.7.2 Processo colaborativo de humanos

A fatoração é consequência da minha *inteligência natural* apoiada por um programa que vem distribuído junto com `Debian/Gnu/linux`, o programa `factor` que tem uma capacidade incrível para fatorar números muito grandes, feito por humanos que apoiam a minha *inteligência natural*. E tenho aqui que agradecer aos programadores que fizeram `factor` que para mim são anônimos, e o trabalho anônimo deles deixando seus programas em domínio público foi essencial para a produção deste artigo como muitos dos textos que eu escrevo.

A fatoração é consequência da minha *inteligência natural* apoiada por um programa que vem distribuído junto com `Debian/Gnu/linux`, o programa `factor` que tem uma capacidade incrível para fatorar números muito grandes, feito por humanos que apoiam a minha *inteligência natural*. E tenho aqui que agradecer aos programadores que fizeram `factor` que para mim são anônimos, e o trabalho anônimo deles deixando seus programas em domínio público foi essencial para a produção deste artigo como muitos dos textos que eu escrevo.

0.8 O programa

A versão para `calc`

```
define DizimaPeriodica(m,periodo,noves, naoperiodo) {
        local denominador = noves;
        local numerador = noves*naoperiodo + periodo;
        if (m ¡ 0) denominador=denominador*power(10,-m);
        if (m ¿ 0) numerador = numerador*power(10,m);
        print numerador,'/',denominador;
return(numerador/denominador);
}
```

Ou a versão para `python3`

```
def Geratriz(m,periodo,noves, naoperiodo):
    denominador = noves;
    numerador = noves*naoperiodo + periodo;
    if (m ¡ 0): denominador=denominador*pow(10,-m);
    if (m ¿ 0): numerador = numerador*pow(10,m);
    print(numerador,'/',denominador);
```

E divirta-se experimentando

- `m = -12 -  0.00012345678914151415`
- `m = -8 -  1.23456789141514151415`
- `m = -4 -  12345.6789141514151415 1415`
- `m = 0 -  123456789.14151415141514151415`
- `m = 4 -  1234567891415.14151415141514151415`
- `m = 8 -  12345678914151415.14151415141514151415`

e naturalmente, testando os programas.

0.8.1 As linguagens de programação

E você pode testar os dois programas na sucessão de exemplos. O programa em `python3` tem que ser digitado <u>exatamente</u> como aparece no texto, em `python3` os *blocos lógicos* são determinados por tabulação e é esta a principal razão porque eu evito esta *magnifica linguagem*, `python3`, porque, com frequência, eu tinha que passar programas para estudantes via `e-mail` e o resultado era desastroso pela perda da tabulação.

A linguagem `calc` segue a sintase da linguagem `C++` em que os blocos lógicos são determinados por chaves. Os programadores, além das chaves, também usam tabulação para aumentar a visibilidade dos blocos lógicos, em `python3` o uso de tabulação não é uma *opção* e sim uma *obrigação* que infelizmente atrapalha quando se muda de editor de textos. Fora esta complicação, `python3` é uma linguagem interpretada, muito rápida, de *precisão infinita* portanto muito boa para se trabalhar com Matemática. `calc` também é uma linguagem de *precisão infinita*.

Cabe acrescentar o que significa *precisão infinita*, nestas linguagens existe um algoritmo interno que nos permite trabalhar com números arbitrariamente grandes. Por exemplo, em ambas é possível calcular fatorial de qualquer número o que é essencial para quem faz pesquisa com números inteiros, em particular com números primos. Penso que `calc` é produto dum grupo americano que faz pesquisas com números primos em que `calc` é excelente.

Índice Remissivo

referências

[Lo11] David I. Bell Landon Curt Noll e other. *Calc - arbitrary precision calculator*. Rel. técn. http://www.isthe.com/chongo/, 2011.

[Pra19a] Tarcisio Praciano-Pereira. *A reengenharia do Cálculo*. Rel. técn. Sobral Matemática, 2019.

[Pra19b] Tarcisio Praciano-Pereira. *A reengenharia do Cálculo, segunda parte*. Rel. técn. Sobral Matemática, 2019.