

Mnemônico para números irracionais Durante milhares de anos, a humanidade desconheceu os números irracionais. (números que não podem ser escritos na forma de uma divisão entre dois inteiros, ou que quando escritos na forma decimal, nunca repetem um padrão). Entretanto eles sempre estiveram por aí. O mais famoso π , é igual a 3,14159265..., sendo os três pontos indicativos de que ele segue ao infinito. Lá no passado, uma maneira de lidar com tais números era achar uma divisão de inteiros que se aproximasse do referido número. Assim, os egípcios usaram por muito tempo a divisão $\frac{22}{7}$ cujo valor é 3.142857143. Mais recentemente, passou-se a usar $\frac{355}{113}$ que é igual a 3.14159292. Outro número irracional importante, o e tem valor 2.718281828... e também é conhecido pelas divisões $\frac{87}{32}$ que é igual a 2.71875 e $\frac{878}{323}$ que é 2.718266254.

Este exercício é simples. Dado um número irracional qualquer, você deve escrever um programa (pode ser no visualg, java, c, c++, ...) que localize os dois inteiros menores do que 100 e outros dois inteiros menores do que 1000 cuja divisão mais se aproxime do número que vocês receberam. Eis a versão do algoritmo feito em visualG:

```
1: algoritmo "num_irrac.alg"
2: var
3: men,dif,x:real
4: i,j,is,js:inteiro
5: inicio
6: men <- 1.0
7: para i de 1 ate 1000 faca
8:   para j de 1 ate 1000 faca
9:     x<- i / j
10:    dif <- abs(x-2.718281828) // ou o numero que se quer
11:    se dif < men entao
12:      is<-i
13:      js<-j
14:      men<-dif
15:    fimse
16:  fimpara
17: fimpara
18: escreva(is,js)
19: fimalgoritmo
```

No caso acima, a sequencia de comandos 9-15 é executada 1.000.000 de vezes. Em um computador com 2 processadores, rodando a 2.8 GHz, com 4 GB de memória (uma baita máquina) o visualG demorou 186 segundos para achar a resposta certa, que é 878,323.

Porém como o número que se busca é maior que UM, só nos interessam as frações cujo numerador é maior que o denominador. Razão pela qual o comando 8 pode ser modificado para

```
8:   para j de 1 ate i faca
```

Note que onde era 1000, agora é i. Este novo programa rodou em 91 segundos tendo feito 500.500 comparações.

PL998b 11/02/2019 12:36:11.7