Soluções dos Problemas



FCUL Rally Pro 2017

1) Números > 10000

É necessário percorrer a lista e contabilizar os números que satisfazem a condição pedida:

```
para cada item elemento na lista Lista de valores 1
faça elemento para 10000
faça definir contador para contador 110000
imprime contador
```

2) O maior elemento menor que 60000

É uma pesquisa pela lista à procura do máximo, mas filtrando números que não satisfazem a condição

```
definir maximo v para
                          na lista 🌘
                                    Lista de valores 2
para cada item (
              elemento *
faça
                   elemento *
                                        60000
      faça
             se
                          elemento *
                                              maximo *
             faça
                          maximo v para
                                            elemento
imprime
           maximo
```

3) Comparar elementos de duas listas

Pesquisar elemento a elemento nas duas listas (a variável indice como índice comum) e contabilizar apenas os pares de elementos que nos interessam.



Uma possibilidade passa por usar o resto da divisão inteira:

```
definir numero v para
definir soma para
repita
       15000
                vez
faça
      definir numero v para
                                numero *
      se
                   resto da divisão de
                                      numero
      faça
            definir soma para
                                     soma *
                                               + *
                                                     numero 1
          soma
imprime
```

Queremos múltiplos de 3 que não são múltiplos de 9.

E queremos os primeiros 10000! Estes são:

3 6 9 12 15 18 21 24 27 30 33 36 39 42 45 48 51...

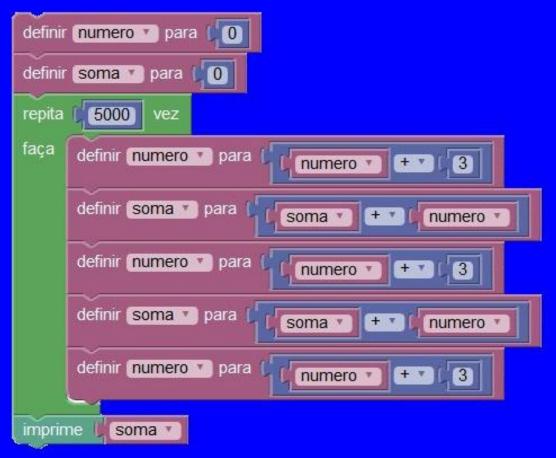
É importante detectar padrões para escrever algoritmos:

os múltiplos de 9 surgem de três em três na lista dos múltiplos de três

Um método: gerar 5000 triplos (15000 números dos quais escolhemos 10000)

O que resulta em num algoritmo como este:

Criamos triplos e um deles não é somado.



Outra possibilidade é fazer umas 'contas' na sequência anterior:

ou seja, podemos somar os números de 1 a 15000 e subtrair os múltiplos de 3 depois é só multiplicar por 3

3 (
$$\sum_{i=1}^{15000} i - 3 \sum_{i=1}^{5000} i$$
) = 3 ($\frac{15000 \times 15001}{2} \times 3 \frac{5000 \times 5001}{2}$)

Usando progressões aritméticas substituimos o algoritmo por uma expressão!

5) Primos entre 1 e 100000

Este problema poderia ser realizando implementando o algoritmo para verificar se um dado número é primo.

No entanto, um programador tem de conhecer as ferramentas disponíveis!

Puzzle: como duplicar a velocidade deste ciclo mudando alguns valores?

```
definir numero para
definir contador para
        99998
repita
                vez
faça
      se
                              é primo
                  numero *
      faça
            definir contador para
                                       contador *
                                                   + *
      definir numero para
                                numero *
          contador
imprime
```



6) Trocos

O truque aqui é começar por subtrair as moedas maiores, e quando o montante é menor que a moeda atual, passar para a moeda seguinte.

Este é um exemplo de algoritmo ganancioso

```
definir moedas_indice v para (11)
repita até *
                montante *
      repita até v
                      montante *
                                   ≥ ▼
                                          na lista
                                                               obter *
                                                                                moedas indice *
                                                   moedas *
            definir moedas indice para
                                            moedas_indice *
      definir montante para
                                 montante *
                                                    na lista
                                                             moedas *
                                                                         obter *
                                                                                   # v
                                                                                          moedas indice *
      definir num moedas * para
                                    num_moedas *
          num moedas
imprime
```



7) Jogo de Bachet

Este problema requer guardar as soluções intermédias para acelerar o cálculo. Usamos uma

Esta técnica é conhecida por programação dinâmica.