

# Colorir Mapas

IP1, [folha6.html](#)

Novembro de 2001

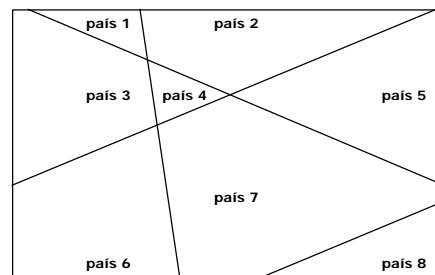
[jlopes@fe.up.pt](mailto:jlopes@fe.up.pt)

## Colorir Mapas

---

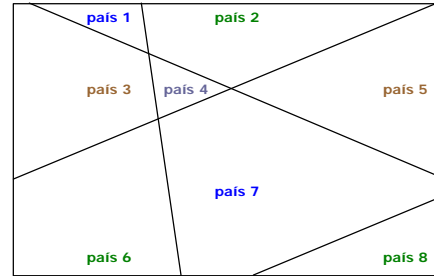
- Pretende-se colorir mapas de países, com um número mínimo de cores e sem que cores iguais sejam utilizadas em países vizinhos.

- São considerados países **vizinhos** os que tiverem:
  - pelo menos, uma linha de fronteira comum
    - caso dos países 1 e 2
  - um ponto de fronteira comum
    - caso dos países 2 e 3



## Abstracção de Dados

- O mapa é modelado através de uma lista, tendo um elemento por cada país do mapa.
  - O elemento correspondente a um país representa os países vizinhos desse país.
  - Por exemplo, o país 1, sendo o primeiro elemento da lista, tem como vizinhos os países 2, 3 e 4.



```
(define mapal
  '((2 3 4)           ; país 1
    (1 3 4 5 7)     ; país 2
    (1 6 7 2 4)     ; país 3
    (1 7 6 2 3 5)   ; país 4
    (7 2 4)         ; país 5
    (3 4 7)         ; país 6
    (2 3 4 5 6 8)   ; país 7
    (7))            ; país 8
)
```

João Correia Lopes  
Faculdade de Engenharia da Universidade do Porto

Colorir Mapas/IP1  
Novembro de 2001

## Abstracção de Dados...

- A modelação das cores é também feita através de uma lista

```
(define paleta-cores
  '(C1 C2 C3 C4)) ; neste caso, utilizam-se 4 cores
```

- A atribuição de cores aos vários países é conseguida com uma chamada a **colorir-mapa**
  - espera dois argumentos, duas listas, uma que representa o mapa e outra o conjunto de cores utilizadas para colorir o mapa
  - devolve uma lista com a indicação das cores a atribuir a cada país, a começar pelo país 1.

```
> (colorir-mapa mapal paleta-cores)
(c1 c2 c3 c4 c3 c2 c1 c2)

> (colorir-mapa mapal '(c1 c2 c3))
nao ha' solucao...
```

João Correia Lopes  
Faculdade de Engenharia da Universidade do Porto

Colorir Mapas/IP1  
Novembro de 2001

## colorir-mapa

---

```
; colorir m apa chama procura-solucao com o m apa, a pale te de cores e um a soluçã o vazia
;
; m apa - lista de países: ((...) (...))
; país - lista de países vizinhos: (1, 2, ... )
; pale te - lista de cores: (c1, c2, ... )
; soluçã o -
;   lista de cores atribuídas a cada país (posiçã o na lista) (c1, c2, c1, ... )
;   não encontra soluçã o: '()'
;
(define colorir-m apa
  (lambda (m apa pale te)
    (procura-solucao m apa pale te '()))
  )
)
```

João Correia Lopes  
Faculdade de Engenharia da Universidade do Porto

Colorir Mapas/IP1  
Novembro de 2001

## procura-solucao

---

```
; procura-soluçã o tenta encontrar um a cor para cada país do m apa
;
;
(define procura-solucao
  (lambda (m apa pale te solucao)
    ;
    (if (null? m apa)
        ; se não há mais países, tem os a soluçã o
        solucao
        ; caso contrário, procura-se um a cor para o m apa, com a pale te e a soluçã o
        ; encontrada até ao m om ento e a pale te de cores que ainda podem ser usadas
        (procura-cor m apa pale te solucao pale te))
    )
  )
)
```

João Correia Lopes  
Faculdade de Engenharia da Universidade do Porto

Colorir Mapas/IP1  
Novembro de 2001

## procura-cor

```
(define procura-cor
  (lambda (m apa pale-te-original solucao pale-te-reduzida)
    (if (null? pale-te-reduzida)
        '() ; se não há mais cores na palete; falha
        (if (cor-boa? (car m apa) (car pale-te-reduzida) solucao)
            (letrec ( ( ; se a cor pode ser usada no país, considerando as cores já atribuídas em solução
                      (solucao-aumentada (append solucao (list (car pale-te-reduzida)))) ; junta cor à solução
                      ; e procura cor para o próximo país (cdr m apa) com a palete com todas as cores
                      (solucao-nova (procura-solucao (cdr m apa) pale-te-original solucao-aumentada)))
                    (if (null? solucao-nova)
                        ; se não encontrou solução passa à próxima cor disponível
                        (procura-cor m apa pale-te-original solucao (cdr pale-te-reduzida))
                        solucao-nova)) ; se encontrou, devolve a solução
            ; se a cor não é boa, tenta com a próxima cor da palete
            (procura-cor m apa pale-te-original solucao (cdr pale-te-reduzida))))))
```

João Correia Lopes  
Faculdade de Engenharia da Universidade do Porto

Colorir Mapas/IP1  
Novembro de 2001

## cor-boa?

*; Um a cor será boa para atribuir a um país se nenhum dos países vizinhos,  
; já com cor atribuída, a utiliza*

```
(define cor-boa?
  (lambda (vizinhos cor solucao)
    (or
      (null? vizinhos) ; a cor serve se não há vizinhos, ou
      (and
        ; serve para este vizinho, e
        (or ; se o vizinho ainda não está na solução
          (< (length solucao) (car vizinhos))
          ; ou a cor não é usada neste vizinho
          (not (equal? cor (list-ref solucao (sub1 (car vizinhos))))))
        ; serve para os próximos vizinhos
        (cor-boa? (cdr vizinhos) cor solucao))))
```

João Correia Lopes  
Faculdade de Engenharia da Universidade do Porto

Colorir Mapas/IP1  
Novembro de 2001