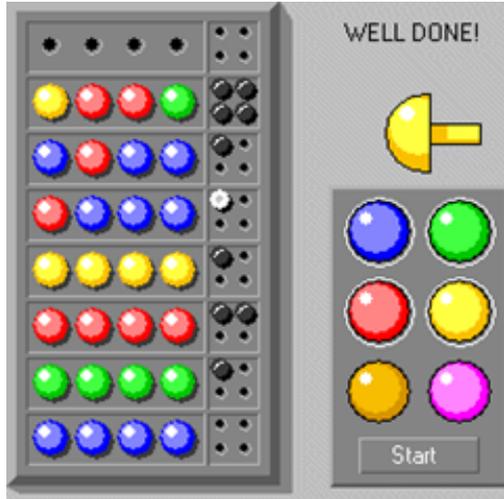


# Code Cracker

(trabalho prático para Introdução à Computação 2005-2006)

Code Cracker é um jogo em que uma pessoa tem de achar o código escondido. O código consiste numa combinação de unidades coloridas. O jogador tenta combinações sucessivas e a cada tentativa recebe informações acerca da combinação,



nomeadamente o número de unidades com cor certa e o número de unidades com a cor e o lugar certos. Por dedução o jogador é capaz de chegar à combinação certa. Um exemplo de um jogo é mostrado aqui ao lado. O jogador começou por tentar a combinação composta por 4 unidades azuis (não é muito inteligente). As informações recebidas foram “nada certo”. A segunda tentativa de todas verdes resultou em “uma unidade no lugar certo”. Depois de mais algumas tentativas, o jogador achou a combinação certa, mostrada na penúltima linha, nomeadamente “amarelo, vermelho, vermelho, verde”.

---

A: Escreva um programa que permite jogar o jogo de Code Cracker. O jogador faz as sucessivas tentativas até achar a combinação final.

---

- O programma é totalmente ASCII; Em vez de cores, o programa usa letras “A”, “B”, “C”, etc.

No início do programa o utilizador tem de especificar

- O número de “cores” e se são permitidas cores repetidas ou não.
- O número de unidades numa linha.
- O número máximo de tentativas.

Depois, o programa tem de

- Gerar um código aleatório a ser determinado pelo jogador.

Um exemplo é mostrado na tabela na página seguinte (O *output* tem que ser mesmo exactamente assim, menos as cores, obviamente!):

---

B: Modifique o programa de forma a incluir a possibilidade do computador jogar contra o utilizador e achar o código por tentivas sucessivas.

---

Um exemplo é dado no lado direito da tabela a seguir. O método sugerido é: Para uma certa linha, gera combinações aleatórias até chegar a uma que é consistente com as informações fornecidas anteriormente (É de notar que este método é muito moroso, especialmente para jogos com um grande número de cores ou colunas).

A	B
<pre> Numero de cores: 6 Cores repetidas (s/n): s Numero de colunas: 4 Numero maximo de tentativas: 8 ----- Codigo gerado tentativa 1: AAAA 1: AAAA: Lugar certo: 0. Cor certa: 0 tentativa 2: BBBB 2: BBBB: Lugar certo: 1. Cor certa: 0 tentativa 3: CCCC 3: CCCC: Lugar certo: 2. Cor certa: 0 tentativa 4: DDDD 4: DDDD: Lugar certo: 1. Cor certa: 0 tentativa 5: CAAA 5: CAAA: Lugar certo: 0. Cor certa: 1 tentativa 6: ACAA 6: ACAA: Lugar certo: 1. Cor certa: 0 tentativa 7: DCCB 7: DCCB: Lugar certo: 4. Cor certa: 0 Ganhou em 7 tentativas! ----- Quer continuar? (s/n): n Obrigado e bom dia </pre>	<pre> Numero de cores: 6 Cores repetidas (s/n): s Numero de colunas: 4 Numero maximo de tentativas: 8 Computador ou utilizador? (c/u): c Escolhe um codigo, &lt;return&gt; quando pronto ----- tentativa 1: AABB Lugar certo: 2 Cor certa: 0 tentativa 2: CABC Lugar certo: 0 Cor certa: 2 tentativa 3: ACDB Lugar certo: 3 Cor certa: 0 tentativa 4: ACEB Lugar certo: 3 Cor certa: 0 tentativa 5: ACFB Lugar certo: 4 Cor certa: 0 Ganhou em 5 tentativas! ----- Quer continuar? (s/n): n Obrigado e bom dia </pre>

Texto normal azul: escrito pelo computador.

**texto itálico vermelho:** escrito pelo utilizador.

### Regras gerais

- Trabalhe em grupos de duas pessoas. Usa a linguagem Turbo PASCAL para o trabalho
- Fim do prazo para entergue: dia 9 de Junho de 2006, 23:59
- Entregue numa disquete 1.4 MB ou via e-mail ([pjotr@ualg.pt](mailto:pjotr@ualg.pt)). Use a conta da universidade. Quando não receber uma mensagem de recepção o trabalho não foi entregue!)
- Escreve os seus nomes, números e e-mail no label da disquete e no início do programa
- Vale 20% da nota final (o exame vale 80%).
- Quem tiver 14 ou mais de 14 tem que vir falar com o docente.
- Se não entregares o trabalho, não serás admitido a exame. Lembra-se que serão admitidos a exame apenas os alunos com 10 ou mais neste trabalho e que tiveram frequência nas aulas práticas.
- **Os estudantes que copiam ou deixaram copiar o seu trabalho não serão admitidos a exame e correm o risco de ser expulsos do curso.**