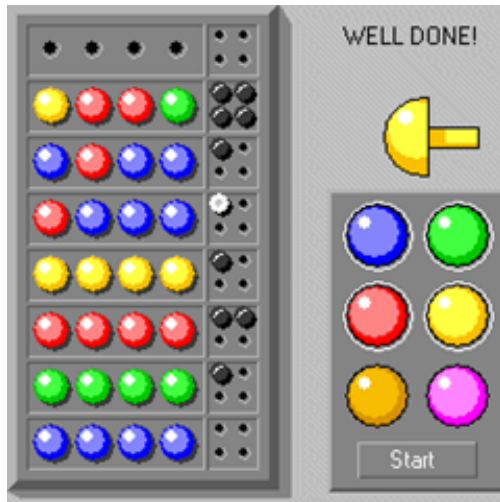


# Code Cracker

(programming exercise Introduction to Programming 2005-2006)

Code Cracker is a game in which a player has to guess the secret code. The code consists of a combination of colored units. The player tries successive combinations of colors and at each trial receives information about the correctness of the combination, namely the number of units that



have the correct color and the number of units that have the correct color and are on the correct place. By deduction, the player is then able to arrive at the correct combination. An example of a game is shown here on the left. The player started with trying a combination of 4 blue units (not very intelligent). The information received was “nothing correct”. A second try of all greens resulted in “one unit on the correct place”. After some more tries, the player managed to find the correct combination, shown on the penultimate row, namely “yellow, red, red, green”.

---

A: Write a program that allows a person to play the game of Code Cracker. The player does the successive guessing until reaching the final solution.

---

- The program is fully text; Instead of colors, the program has to use letters “A”, “B”, “C”, etc.

In the beginning the program has to ask

- The number of “colors” and the possibility of colors repeated or not.
- The number of units in a row.
- The number of allowed trials.

The program then has to

- randomly generate a code to be cracked by the player.

An example is given in the left side of the table on the next page (the output has to be exactly like this, without the colors, of course!):

---

B: Modify the program to include the possibility that the computer plays against the user. The computer has to try to find the combination by successive trial-and-error.

---

An example of this is given in the right side of the table on the next page. Method suggested: For a certain turn, generate combinations until a possible one comes out. (This method can be time consuming though, especially with a large number of colors or columns).

A	B
<pre> Number of colors: 6 Repeated colors (y/n): y Number of columns: 4 Maximum number of tries: 8 ----- Code generated try 1: AAAA 1: AAAA: Correct place: 0. Correct color: 0 try 2: BBBB 2: BBBB: Correct place: 1. Correct color: 0 try 3: CCCC 3: CCCC: Correct place: 2. Correct color: 0 try 4: DDDD 4: DDDD: Correct place: 1. Correct color: 0 try 5: CAAA 5: CAAA: Correct place: 0. Correct color: 1 try 6: ACAA 6: ACAA: Correct place: 1. Correct color: 0 try 7: DCCB 7: DCCB: Correct place: 4. Correct color: 0 You have won in 7 times! ----- Do you want to try again? (y/n): n Thank you and have a nice day </pre>	<pre> Number of colors: 6 Repeated colors (y/n): y Number of columns: 4 Maximum number of tries: 8 Computer or Player? (c/p): c Think of a code, &lt;return&gt; when ready ----- try 1: AABB correct place: 2 correct color: 0 try 2: CABC correct place: 0 correct color: 2 try 3: ACDB correct place: 3 correct color: 0 try 4: ACEB correct place: 3 correct color: 0 try 5: ACFB correct place: 4 correct color: 0 I have won in 5 times! ----- Do you want to try again? (y/n): n Thank you and have a nice day </pre>

Blue normal text: written by computer.

*Red bold italics*: written by user.

### General rules

- Work in groups of two (2) people. Use Turbo PASCAL for your program.
- Deadline: 9 June 2006, 23:59.
- Hand in diskette 1.4 MB or send e-mail ([pjotr@ualg.pt](mailto:pjotr@ualg.pt)). Use your university account. If you don't receive an acknowledgement message, your program was not received!).
- Write your names on the disk and in the beginning of the program.
- Weight for final mark 20% (the exam weighs 80%).
- Who has 14 or higher has to defend the work with the docent.
- Without handing in the program, you will not be admitted to the exam. Remember, you are only admitted to the exam if you have 10 or higher for the practical work and was present at 67% of the practical lectures.
- **Students that copy or let copy their work are not admitted to the exam and possibly are expelled from the course.**