Course C^{++}, Exercise Number 7

05.04.2012 + two weeks

First have a look at Task 8 from the ANSIC Class. We are going to make this task in C^{++} in a decent style. Download the files **assert.h, square.h, square.cpp**.

1. Complete the missing methods of **class square**.

2. The **class guess** assigns an element in the square, which will be withdrawn when the guess is destroyed. In this way, one can write code like

   ```
   for( unsigned int k = 0; k < sq. getdimension( ); ++ k )
   {
      guess g = square::guess( sq, position(4,5), k );


   } // Guess automatically backtracked. Don't call a
     // destructor by yourself.
   ```

   In this small example, it is not really worth the effort, but it is useful to know about this technique.

3. Complete the programm. Put the table on Page 2 of the C exercise in a vector.

   ```
   std::vector< position > order;
   order. push_back( position(0,0) );
   order. push_back( position(.... ));
   ```

   Write the function

   ```
   void printmagicsquares( unsigned int level,
                           square& sq,
                           const std::vector< position > & order );
      // Do not consider making sq or order a global variable!
   ```

(I will be more impressed if you manage to find 5×5 squares. In order to do this, you have to find a good guessing strategy.)