Exercise Sheet #5

Bulcsú Sándor <sandor@th.physik.uni-frankfurt.de>
Hendrik Wernecke <wernecke@th.physik.uni-frankfurt.de>
Laura Martin <lmartin@th.physik.uni-frankfurt.de>
Christopher Czaban <czaban@th.physik.uni-frankfurt.de>

Problem 1 (*Control flows 1*) 12 Pts

(a) Write a program that replaces every second element of an array by zero, using a `for` loop and iterating with step size two. (2 Pts)

(b) Write a program that prints out all odd numbers smaller than 20 by using a loop and the `continue` statements. (2 Pts)

(c) Write a program that calculates the largest factorial smaller than a given upper limit. Therefore use a loop and the `break` statement to exit, when the next number is larger than the limit. (2 Pts)

(d) Binomial coefficient (2 Pts)
Write a program that computes the binomial coefficient $\binom{n}{k}$ where n and k are specified by the user.

(e) Password prompt (2 Pts)
Write a program that asks the user to enter a password. The user may only enter the password a specified number of times. The program should check the password every time and exit if the correct password was given or the maximal limit is reached.

(f) The guessing game (2 Pts)
Write program that randomly generates a number between 1 and 100 and the user has to guess this number. If the user guesses too high/low the program outputs "too high/low" until the user hits the correct number.

Problem 2 (*Go*) 8 Pts

This year progress in AI enabled computers to beat a human champion in the famous board game `Go`. The rules are pretty simple: Two players, Black and White, take turns placing a stone (game piece) of their own color on a vacant (intersection) point of the grid on a Go board. Vertically and horizontally adjacent stones of the same color form a chain that cannot subsequently be

subdivided and, effectively, becomes a single larger stone. A vacant point adjacent to a stone is called a liberty for that stone. Stones in a chain share their liberties. A chain of stones must have at least one liberty to remain on the board. When a chain is surrounded by opposing stones so that it has no liberties, it is captured and removed from the board. The goal is to capture as many of the opponent's stones as possible. (Source: `https://en.wikipedia.org/wiki/Go_(game)#Basic_rules`)

Implement here a modified version of the game according to following instructions.

(a) Create a 7×7 Go board and randomly place 10 stones of each color on the grid. Stones should not overlap.

(b) Write a method that simulates the move of two players having different strategies. One player places his stones such that there is at least one of his stone on an adjacent point. The other player places his stones on one of the fields with the highest number of opponent's stones on the neighbour points, but less then four. Hint: ignore special cases for the spots at the border.

(c) Print the board with its stones after the initial randomization was generated and as well after the 20 moves have been played. You can use $'o'$ for white and $'\#'$ for black stones.

Here two examples for a 7×7 gird after a randomized initial set (left) and after

	1	2	3	4	5	6
1				o		
2	o		#			
3	o	#				o
4			#			
5				#	#	
6		o				

	1	2	3	4	5	6
1	#		o	o	#	o
2	o		#		#	o
3	o	#		o	o	o
4	#		#		#	o
5	o	#		#	#	o
6	o	o				o