# Software lab summer term 2016
# Implementation of Heuristic Algorithms for Board Games
# – Assignment 4 –

**Next meeting is on 21.06.2016. Upload your code and report before the deadline of 24.06.2016.**

## Task 1

Extend your implementation with a move sorting algorithm which can be switched off. The idea of move sorting is to explore "promising" branches in the search tree first, such that the search space becomes smaller as $\alpha$ and $\beta$ converge faster.

## Task 2

Analyse how the search efficiency is enhanced by move sorting. Therefore evaluate the performance parameters (cf. Task 3.3) on at least 10 different maps. Use tables and graphs to visualise your data.

## Task 3

Implement the iterative deepening search algorithm. It performs a depth-first search beginning with depth one and increases its depth to $(i+1)$ when all nodes on level $i$ have been explored. This procedure is repeated until a (user defined) depth $d$ has been reached.

In a second step, extend your iterative deepening algorithm with a time deadline. This means that the search should stop after a given a deadline of $t$ time units is reached.

Think of a useful value that your time limited search algorithm shall return.

**Note that on the 24.06 we expect your client to be fully functional. It must be able to play without errors, and MiniMax, Alpha-Beta, move sorting and iterative deepening must be implemented correctly. Again, this is a strict deadline.**