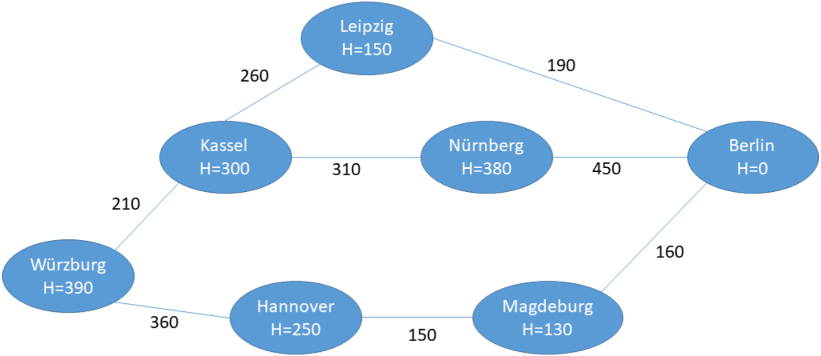
Artificial Intelligence

Exercise Sheet 2

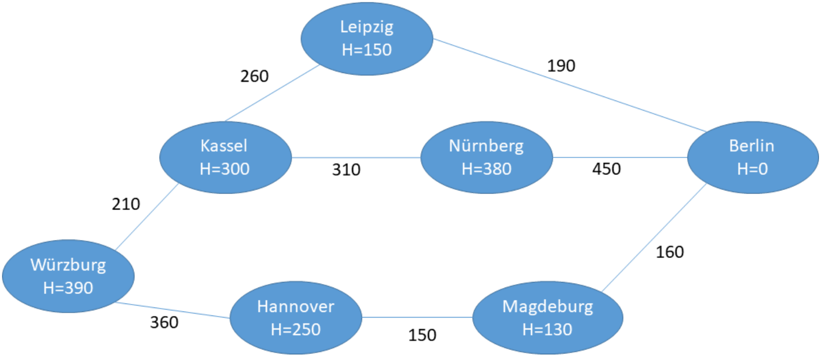
November 2, 2021

Task 1



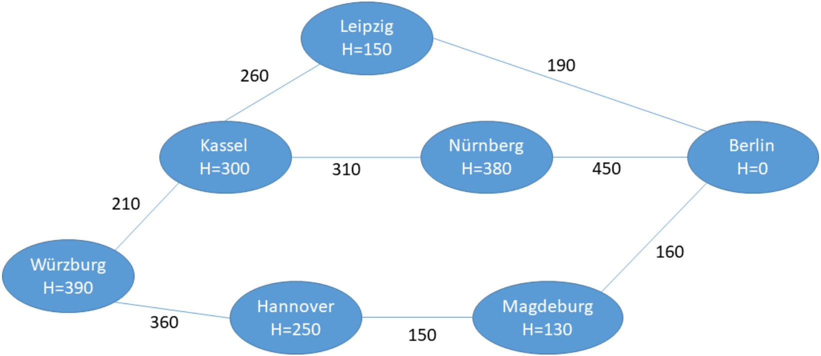
Expanded	Queue

Task 1



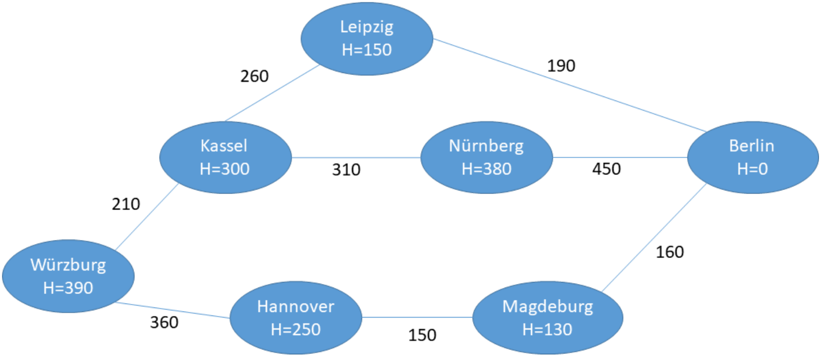
Expanded	Queue
W	K (210 + 300), H (360 + 250)

Task 1



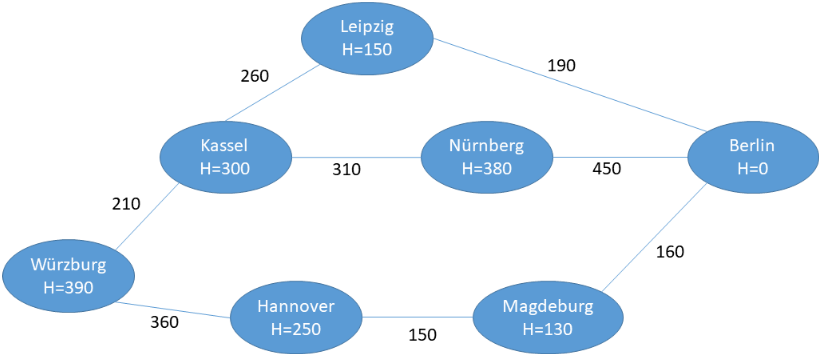
Expanded	Queue
W	K (210 + 300), H (360 + 250)
K	H (360 + 250), L (470 + 150), N (520 + 380)

Task 1



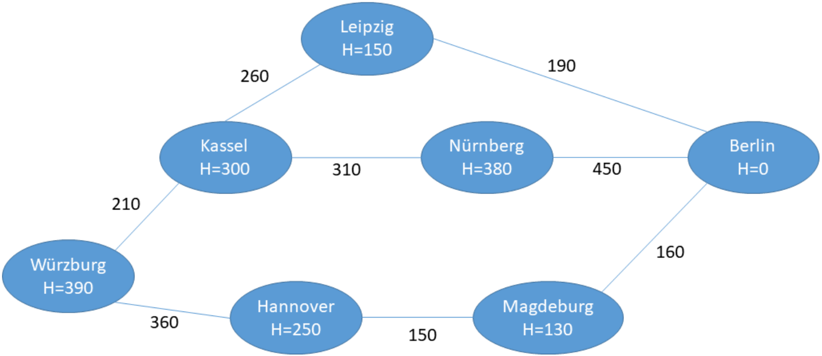
Expanded	Queue
W	K (210 + 300), H (360 + 250)
K	H (360 + 250), L (470 + 150), N (520 + 380)
H	L (470 + 150), N (520 + 380), M (510 + 130)

Task 1



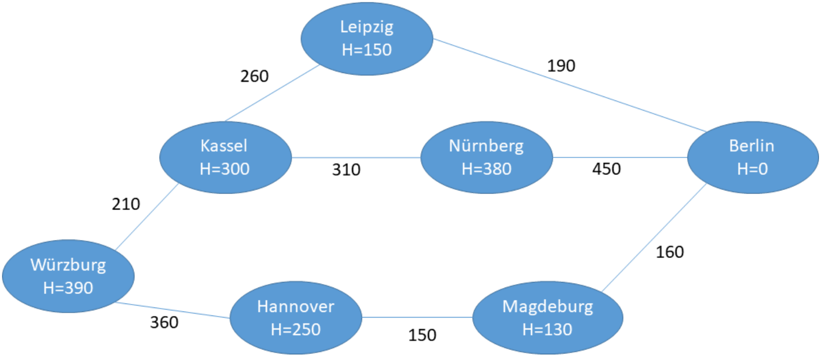
Expanded	Queue
W	K (210 + 300), H (360 + 250)
K	H (360 + 250), L (470 + 150), N (520 + 380)
H	L (470 + 150), N (520 + 380), M (510 + 130)
L	N (520 + 380), M (510 + 130), B (660 + 0)

Task 1



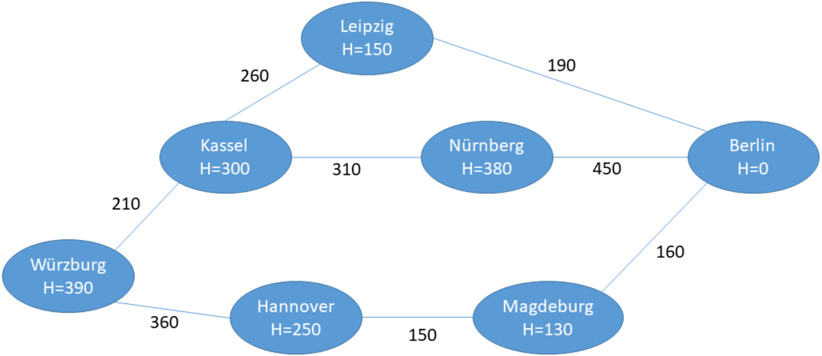
Expanded	Queue
W	K (210 + 300), H (360 + 250)
K	H (360 + 250), L (470 + 150), N (520 + 380)
H	L (470 + 150), N (520 + 380), M (510 + 130)
L	N (520 + 380), M (510 + 130), B (660 + 0)
M	N (520 + 380), B (660 + 0)

Task 1



Expanded	Queue
W	$\underline{\text{K (210 + 300)}}$, $\underline{\text{H (360 + 250)}}$
K	H (360 + 250), $\underline{\text{L (470 + 150)}}$, $\underline{\text{N (520 + 380)}}$
H	L (470 + 150), N (520 + 380), $\underline{\text{M (510 + 130)}}$
L	N (520 + 380), M (510 + 130), $\underline{\text{B (660 + 0)}}$
M	N (520 + 380), B (660 + 0)
B	

Task 1



Expanded	Queue
W	K (210 + 300), H (360 + 250)
K	H (360 + 250), L (470 + 150), N (520 + 380)
H	L (470 + 150), N (520 + 380), M (510 + 130)
L	N (520 + 380), M (510 + 130), B (660 + 0)
M	N (520 + 380), B (660 + 0)
B	

W -> K -> L -> B

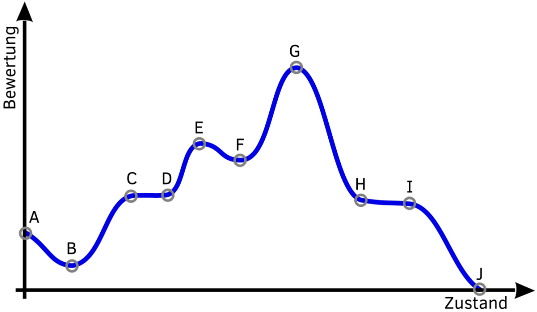
Task 2.1

Pseudo-Code for hillclimbing

```
function Hill-Climbing(problem):
    current ← getInitialState(problem)
    while true do:
        children ← getChildren(current)
        next ← getStateWithHighestValue(children)
        if next.value ≤ current.value then
            return current.State
        current ← next
```

Task 2.2

State the Problems and end points for the following starting points



Start	End	Problem	Solution
B	A/C	undetermined Direction	no
C	C	Plateaus	allow side-steps
D	E	local maxima	allow worsening (Simulated Annealing)

Task 2.3

Solution for local maxima / minima

- Start at multiple different points (Beam-Search)
- Random Restart

Task 3

State a possible scoring function

Number of queens that cannot take another queen

State the transitions between one and another state Move a Queen to an arbitrary free square

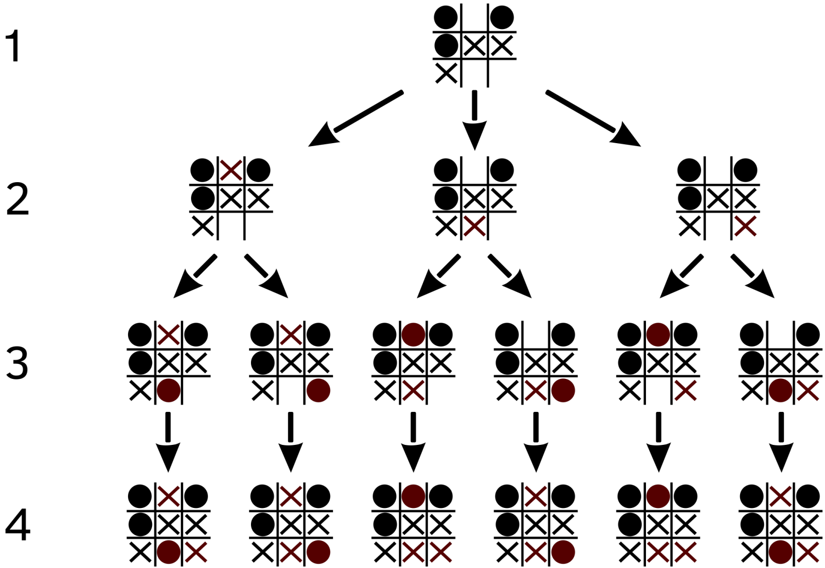
...Improving the Search

Assumption: Starting point contains only one queen for each Row / Column

→ Swap two columns / Rows

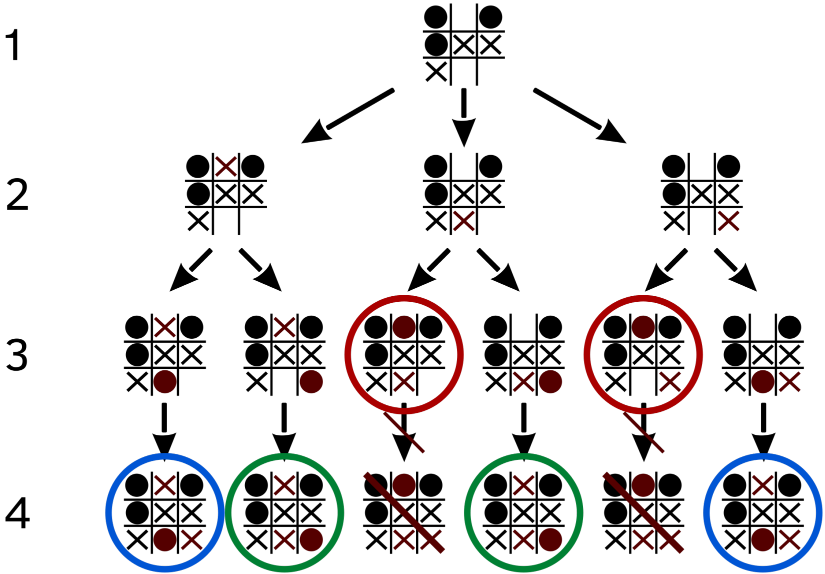
Task 4.1

Complete the Figure



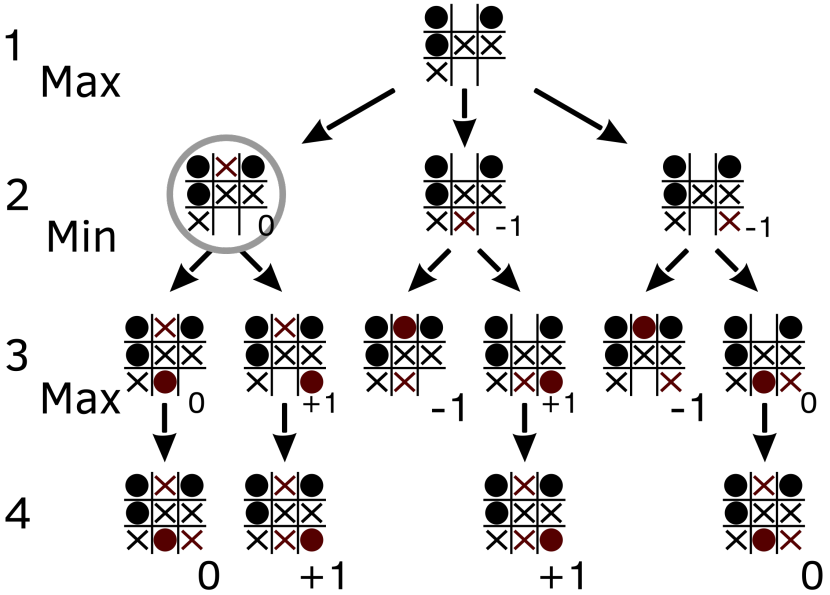
Task 4.2

Define the winning / losing / tied states



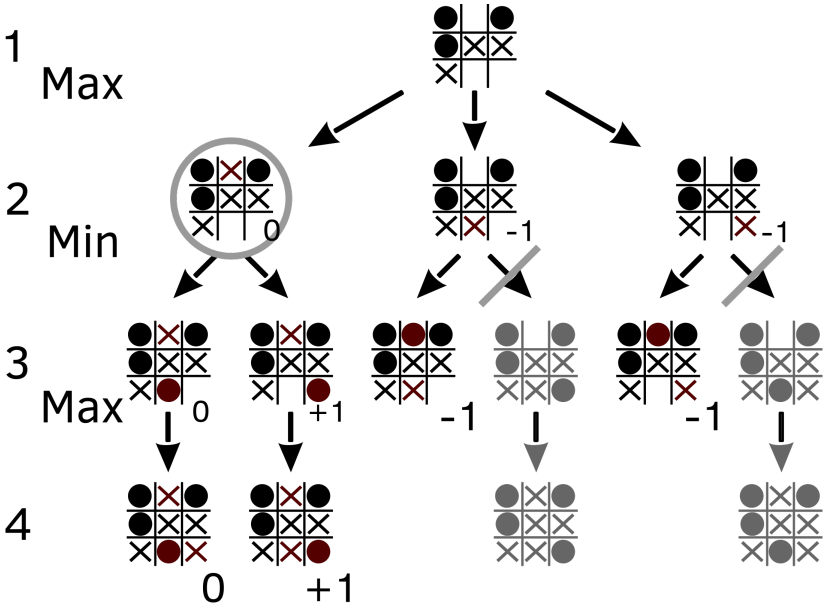
Task 4.3

Min-Max



Task 4.4

Alpha-Beta



Task 4.5

Min-Max for trees that are to deep

By using a heuristic that estimates the evaluation for the respective state and then calculates with this value.

Task 5

Pruning in Expecti-Min-Max

If the possible rating for one position lies in a interval it is possible to estimate how much the average value can change. With this information a pruning is possible.