# CS 404 – Artificial Intelligence
# HW 2 – Blind Search – AIMA– Chp. 3

75pt
Late homeworks accepted for 2 days (no penalty in the first late day; -10pts off when late for 2 days)

**Please type your answers and use only the allocated space.**
**You may color your answers blue for easy grading.**

**Objective:** To deepend the understanding of time and space complexity in search algorithms and deciding on suitable algorithms for a given problem.

**Type your answers, but you can draw any illustrations by hand** (if so you can send the scanned document).

**1) 30pts –Answer the following using the general Tree Search algorithm** (remove front node from the fringe/queue – goal test – expand).

Reminder: You can use the following equality for compactness:

$1 + b + b^2 + \dots b^d = (b^{d+1}-1)/(b-1)$

**a) 15pt** - How many nodes are ***visited*** (chosen from the queue, goal tested and expanded) in the worst case using Breadth-First search, when the solution is at depth d, and the branching factor is b, and the depth of the maximum branch is m?
Give a formula.

level $0$ → $1$ node
$1st$ level → $b$ nodes
$2nd$ level → $b^2$ nodes
…
level $d$ → $b^d$ nodes

In the worst case, we look at $\mathbf{1 + b + b^2 + \dots + b^d}$ nodes, thus, the answer is $\frac{\mathbf{b^{d+1}-1}}{\mathbf{b-1}}$.

**b) 15pt-** How many nodes are ***generated*** (added to the queue as a result of expanding the parent) in the worst case using Breadth-First search, when the solution is at depth d, and the branching factor is b, and the depth of the maximum branch is m?

We add all the remaining nodes and their children except that for the goal node, thus:

$$\frac{b^{d+2}-1}{b-1} - b$$

**2) 45pt – You are given the problem of finding whether 6-degrees of separation holds between a particular 2 people in the world.** E.g. given two people – say you and your favorite celebrity - the software should decide whether they are connected in at most 6 friendship edges (e.g. you-f1-f2-f3-f4-f5-celebrity).

**Let`s assume you have the list of all friendships for all people in the world and that everyone has exactly b=100 friends and that there are 6 billion people in the world.**

a) **18pts)** State **whether the following algorithms are <u>complete</u>** (if there is a up to 6-degree path, does it find it?) **and <u>optimal</u>** (defined here as 'does it find the shortest path connecting two people') **for this problem**.

b) **12pts) If an algorithm is BOTH complete AND optimal, comment on its time and space complexity with a one line summary about its suitability (e.g. "will take too much time/space: O(b^d)" ).** If an algorithm would take too much time or space to be feasible, indicate as such; if it is suitable but is an overkill, you should indicate that also.

| Algorithm | **Complete (answer as Yes or No)** | **Optimal (answer as Yes or No)** | **Feasibility (add a one line comment)** |
|---|---|---|---|
| Breadth first search | Yes | Yes | No, because its time and space complexity are large.<br>Time: $O(b^d)$<br>Space: $O(b^d)$ |
| Depth first search without repeated state checking | No | No | No<br>Time: $O(b^m)$<br>Space: O(bm) |
| Depth first search with repeated state checking | Yes | No | No, the depth could be too large<br>Time: $O(b^m)$<br>Space: O(bm) |
| Depth limited search DFS with a depth limit of ……6…… | Yes | No | Yes, since with m=6, time complexity $O(b^6)$ and space complexity $O(6b)$ |

| | | | |
|---|---|---|---|
| Iterative deepening DFS | Yes | Yes | Yes. Its time complexity is $O(b^d)$ and has a space complexity of $O(bd)$, which is better than BFS. |
| Bidirectional search | Yes | Yes | It is the best uninformed search algorithm with both time and space complexity $O(b^{\frac{d}{2}})$. However, it may not be applicable because the objective problem is known, and thus, is not feasible. |

**c) 15pts) Which blind search algorithm** (among the ones listed above**) would be best for this problem? Explain your answer.** Consider space, time complexities and completeness and optimality.

If two algorithms are the same or similar, you may choose the one which is easier to implement or state that they are both as good / suitable.

Both breadth first search and bidirectional search could be considered for this problem, since they are both complete and optimal.

However, while having the same time complexities, the iterative deepening DFS is better than both of them regarding space complexity, which is $O(bd)$. Moreover, bidirectional search could become harder to implement when operators are not reversible. Thus, iterative deepening DFS would be the best for this purpose.