Heuristic Optimisation

Problem sheet 3

1. Consider the following problem:

   There are given a finite number of points in the plane $P_1, \dots, P_n$, where P_i has coordinates x_i, y_i. The task is to find the distance between the pair of points P_j, P_k that are the closest to each other.

   (a) Discuss how exhaustive search could be applied to this problem.

   (b) Devise a divide and conquer algorithm to solve this problem.

2. Consider the following Boolean satisfiability problem

$$\bigwedge_{k=1}^{n} \left[\left(x_{3k-2} \bigvee x_{3k-1} \right) \bigwedge x_{3k-1} \bigwedge \left(\overline{x}_{3k-2} \bigvee \overline{x}_{3k-1} \right) \bigwedge \left(\overline{x}_{3k-1} \bigvee x_{3k} \right) \right],$$

   where $n \geq 2$ is an integer and we denote

$$\bigwedge_{k=1}^{n} H_k = H_1 \bigwedge H_2 \bigwedge \cdots \bigwedge H_n.$$

   for any Boolean functions $H_1, H_2, \dots, H_n$.

   (a) What is the size of the search space? Justify your answer.

   (b) Is it possible to use the divide and conquer method to solve this problem? Justify your answer.

   (c) Solve the problem.

   (d) Assume that we replace x_{3k} in the problem with x_{3k+1}, without making any other changes. Is it possible to use the divide and conquer method to solve this modified problem?

3. The 8-puzzle is a sliding puzzle that consists of a frame of numbered square tiles with one tile missing. The goal of the puzzle is to get from a **Start state** to a **Goal state** by making sliding moves that use the empty space.

   Show the search tree/graph expanded by best first search for the following 8-puzzle problem and using the "tiles in wrong position" heuristic:

   Start state

	2	3
1	4	5
8	7	6

   Goal state

1	2	3
8		4
7	6	5

4. In some cases there is no good evaluation function for a problem, but there is a good comparison method, which tells whether one node is better than the other, without assigning numerical values to either. Explain whether it is possible to do best-first search based on this comparison, without a proper evaluation function. Is it possible to apply A* search using this comparison?