Paul Lintilhac

Artificial Intelligence – for Professor Davis

Problem 1:

1. If we are to characterize the scheduling problem as a tree-structured state space search problem, then we could consider each level i of the tree to represent task i, with the state being which process the task has been assigned to. So each node i of the tree has exactly two children, representing whether the next (i+1)-th task is given to process 1 or process 2. Therefore the operator is the switching of a task from task 1 to task 2, and the branching factor is exactly 2. The depth of the goal node is then always going to be 4, since we need to assign every task to a process.
2. Below is an example of a portion of the state space for a depth-first search.

T­4: P1

T­3: P1

T­4: P1

T­3: P2

T­2: P1

T­1: P1

ø

T­4: P2

1. Below is an example of the state history for a breadth-first search

T­2: P1

T­1: P2

T­1: P1

ø

T­2: P2

Problem 2:

1. In this scenario, the states are the same as well as the operator, and the same branching factor. The main difference is that the goal state is now different. Now the constraint that all task be assigned to a process has been lifted, and in its place we have a constraint on the total value of all tasks. Therefore the depth of the goal state is no longer set: it could be less than 4 if that satisfies both the deadline and total value criteria.
2. Note that in general there is no difference between the state spaces that will be searched in this case as compared to problem 2, because the operator, states, and branching factor are all the same. The only difference is that the algorithm may now terminate at a level < 4.

T­4: P1

T­3: P1

T­4: P1

T­3: P2

T­2: P1

T­1: P1

ø

T­4: P2

1. Again there is no difference in the state history for problem 2 relative to problem 1 the only difference is that the goal state is now different.

T­2: P1

T­1: P2

T­1: P1

ø

T­2: P2