Some sample questions are given for each session of the module. The exam questions cover all content in the lectures.

1. Constraint Satisfaction Problems

a) Formally define Constraints in constraint satisfaction problems.

A constraint $C_{i,j,k}...$ is a subset of possible relations between values of variables $x_i, x_j, x_k, ...$; $C_{i,j,k}...$ is a subset of $D_i \times D_j \times D_k \times ...$

A constraint in a set of variables defines a restriction on the values the variables can simultaneously take. That is to say what values are allowed, or disallowed among the variables.

b) Define Solution in constraint satisfaction problems.

A solution to a CSP is an assignment of each variable with a value (within its domain), such that all constraints $C$ are satisfied.

2. Search tree and systematic search

a) Explain how Generate and Test is implemented.

At each step, a node in the search tree is generated by labelling a variable in the problem. That is to assign a value to a variable in the CSP. If the label of the node is consistent with all the previous labels, that is satisfying the constraints on the related variables, then this label is tested and accepted. A branch in the search tree is taken, leading to a child node of the current node in the search tree.

This process is repeated until a complete assignment (solution) is found, or a dead-end is reached. In this case, no valid value can be found for the variable, the GT backtracks to the last variable assigned. In the search tree, this is implemented as go back to the parent node and assign a new valid value to the variable. So a different branch will be taken, leading to a continuous search.

3. Consistency techniques

a) What do arc consistency algorithms do? Explain the aims of arc consistency algorithms.

What the arc consistency algorithms do is to repeatedly restrict the domains of the variables until the property hold true. That is all the arcs are made consistent. They examine each arc in the constraint graphs in turn, and remove those values of the first node that are not consistent with any values of the second variable. If any value from the domain of one variable do not have any value of another variable which satisfies the constraint, these values are removed.

The aim of the arc consistency algorithms are that the size of the CSP is reduced so as easier to solve. The earlier this is made, the smaller the size of the search tree is, thus the more efficient the search is to find solutions quickly. The reduced CSPs after applying arc-consistency should be equivalent to the original one. That is, no solutions should be ruled out by the algorithms.

c) Present the pseudo-code of AC-1 algorithm.

PROCEDURE AC-1(Z, D, C)
BEGIN
    Achieve node consistency
    Construct the constraint list Q
    REPEAT
        Changed $\leftarrow$ False;
        FOR each item in Q DO
            Changed $\leftarrow$ Revise($x \rightarrow y$, (Z,D,C))
        UNTIL NOT Changed
    Return (Z, D, C)
d) What is the difference between AC-1 and AC-3 algorithm? (5 marks)

AC-1 checks all constraints whenever any values are removed from the domain of any variables in the previous iteration. This includes those even not affected by the removal of values in the previous iteration. (2 marks)

AC-3 only checks those variables which are affected by the variables whose domains are changed by the removal of values in previous iterations. (2 marks)

AC-3 is more efficient than AC-1. (1 mark)

4. Modelling

a) Explain how symmetry can be avoided for CSPs. (4 marks)

There are some approaches investigated in the research to break symmetries in CP. Among them, re-formulation is the most effective way. However there is no general technique in the field. (2 marks)

Beside re-formulation there are also other approaches. This includes adapting the search algorithm to break symmetry; in addition to the constraints stated in the problem, additional constraints could be added to destroy symmetries; or to build the search tree so that no symmetry arises. (2 marks)

5. Ordering

a) Define the bandwidth of a node and an ordering for constraint graphs. Find the bandwidth of the below ordering. Explain how you obtain this.

The width of the ordering is 3, calculated as below. (2 marks)
Bandwidth of a node is the maximal distance between this node and other connected nodes in the ordering. Bandwidth of an ordering is the maximal bandwidth of all the nodes in the ordering. (4 marks)

The bandwidth of all the nodes in the below ordering is calculated, then the maximal width of them which is 3 is the bandwidth of the ordering. (2 marks)

b) What is the aim of finding the minimal bandwidth of an ordering of variables? (4 marks)

The aim of finding the minimal bandwidth of an ordering is to reduce the distance of backtracks when they are necessary on the search. The smaller the bandwidth of an ordering, the less distance the backtracking need to go back in the search tree. (4 marks)

c) Present and explain an example of heuristic that may be used for variable ordering. (8 marks)

A general heuristic is variable ordering is to label the most constrained variables earlier, so that they won’t be left in the later stage of the search where less number of values is left for these constrained variables. (4 marks)

Largest degree heuristic is widely used in CP. Degree of a node is defined as the number of adjacent edges to the node. So the larger the degree of a node is, the more constrained the node is in the constraint graph. Largest degree heuristic orders the variables decreasingly on the degree of nodes in the graph. (4 marks. Marks are also given for any other variable ordering heuristics introduced in the course, or in the literature.)
6. Constraint optimisation problem

a) Explain how Brunch and Bound work. (8 marks)

B&B is based on depth first search on the search tree for CSPs. In B&B, there are two important factors: a heuristic function and a bound. (2 marks)
The heuristic function is used to estimate the lower or upper bound value of compound values of partial solution during the search. It maps the partial solutions to a numeric value. Good heuristics reduce the search tree to be explored and speed up the search. (3 marks)
The bound is used to indicate the best solution obtained so far, and is updated to cut-off branches which have no better solutions of better value, and thus no optimal solutions. During the search if a better partial solution is obtained with better objective value, the bound will be updated. (3 marks)

7. Others

a) Explain the difference between Backward Checking and Forward Checking. (6 marks)

Both backward checking and forward checking remove the in-consistent values from the variables. Forward checking removes the values of un-initiated variables which are in-consistent with the current assignment. If in any case one the constrained variables has empty domain, the current value is discarded for the current variable. (3 marks)
Backward checking remembers the in-compatible value pairs after met a failure. Then later during the search as long as one of the values in the value pair is assigned to the variable, the other value won’t be attempted for assigning the other variable. As this happens when a failure is met, backtrackings are needed. Backward checking is thus inferior to forward checking. (3 marks)