UNIVERSIDADE DE COIMBRA
Faculty of Science and Technology
Department of Informatics Engineering

# Laboratório de Programação Avançada

Second Written Test – May 31 2017

Name: _____ Student ID: _____

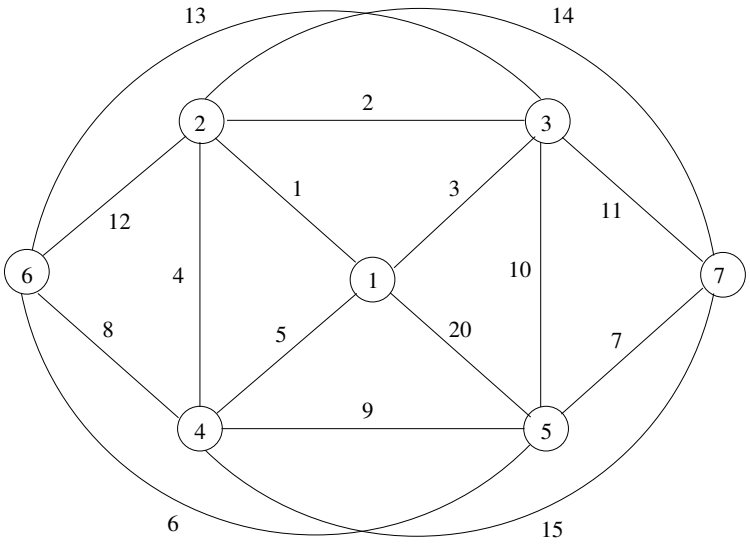
4 grade points in total, 1 hour and 30 minutes, closed books.

1. Given a set of $n$ objects, each of which with a value $v_i$, weight $w_i$ and a volume $z_i$, $i = 1,\ldots,n$, and a knapsack with a maximum weight capacity $W$ and a maximum volume capacity $Z$, the goal is to select a subset of objects that maximizes the total value and does not exceed both the weight and the volume constraint. Consider the following pseudo-code of a backtracking approach for this problem.

```
Function knapsack(i, v, w, z)
1: if w > W or z > Z then
2:     return
3: if i = n then
4:     if best < v then
5:         best = v
6:     return
7: knapsack(i + 1, v, w, z)
8: knapsack(i + 1, v + v_i, w + w_i, z + z_i)
9: return
```

In the box above, write a branch-and-bound approach to solve this problem by adding only the required lines to perform the pruning test. In the box below, provide a justification for the correctedness of your pruning test. The grade to this answer depends on the tightness of the pruning test. (1 g.p.)

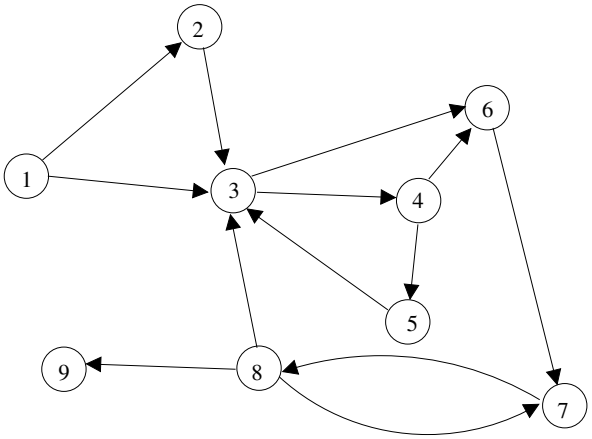
2. Find the minimum spanning tree in the following network using Kruskal algorithm. Design the minimum spanning tree as well as the graph of the union-find data structure (without path compression) in the two boxes below. Always connect the root of the tree with the smallest height to the root of the tree with largest height and, in case of a tie, choose as root the node with the smallest label. (1 g.p.)




Minimum spanning tree:

Union-find data structure:

3. Find the strongly connected components in the following directed graph using Tarjan algorithm. Report the DFS tree(s) starting from node 3 and choosing the nodes for traversal in increasing order of the labels. In case you need more than one DFS tree, start from the node available with the smallest label. Explicitly indicate the nodes that belong to each strongly connected component as well as the final values for `dfs` and `low` at each node.(1 g.p.)



4. Given a spanning tree $T = (V, E)$, where $V$ is the set of vertices and $E$ is the set of edges, sketch the pseudo-code of an algorithm that sequentially deletes all the vertices of this tree such that each deletion leaves the remaining spanning tree connected. In addition, explain it briefly and provide its time complexity. The grade to this answer depends on the efficiency of your approach in terms of time complexity. (1 g.p.)