OPERATIONAL RESEARCH 1 EXAM (Prof. Fischetti) – THEME A

1) Theory

1a) Prove the equivalence between vertices and basic feasible solutions in Linear Programming

1b) Prove the correctness of the Prim-Dijkstra algorithm for the Shortest Spanning Tree problem

1c) Write an Integer Linear Programming model for the Steiner tree problem

2) Linear Programming: Solve the LP problem associated with the following tableau:

0	-3	-2	0	0
3	2	-3	1	2
4	3	2	-1	1

Use the two-phase method, Bland's rule. In the end, report the optimal solution identified (or "impossible" or "unlimited" depending on the case).

3) Integer Linear Programming: solve the ILP problem associated with the following tableau, using Gomory's cutting plane algorithm. At each iteration, add to the current tableau the Gomory cut from the generating row with minimum index (skipping the row associated with the objective function), and reoptimize the tableau with the dual simplex algorithm (Bland's rule). Report all the tableaux obtained, highlighting the pivot element with a circle.

-z	4	0	0	0	1
x3	1/3	0	-13/3	1	4/3
x1	4/3	1	2/3	0	1/3

4) Graph Theory: Write an Integer Linear Programming model for the following problem. Let G=(V,A) be an assigned directed graph in which the arcset A is partitioned into two assigned sets A1 and A2. For each (i,j) in A, a cost c(i,j) and a weight w(i,j) are assigned. You want to find a Hamiltonian circuit of minimum overall cost (travelling salesman problem) that satisfies the following additional constraints:

(i) the total cost of the zero-weight arcs chosen in A1 shall not be less than the total cost of the negative-weight arcs chosen in A2.

(ii) the average weight of the arcs chosen in A2 shall not exceed an assigned value M.

SOLUTIONS

LINEAR PROGRAMMING

```
PHASE 1
  +----------------------------------------------------------------------------
-z|   -7   |  -5      1      0     -3      0      0
  +----------------------------------------------------------------------------
x5|    3   |   2     -3      1      2      1      0
x6|    4   |   3*     2     -1      1      0      1
  +----------------------------------------------------------------------------

  +----------------------------------------------------------------------------
-z|  -1/3  |   0    13/3   -5/3   -4/3     0     5/3
  +----------------------------------------------------------------------------
x5|   1/3  |   0   -13/3    5/3*   4/3     1    -2/3
x1|   4/3  |   1     2/3   -1/3    1/3     0     1/3
  +----------------------------------------------------------------------------

  +----------------------------------------------------------------------------
-z|    0   |   0      0      0      0      1      1
  +----------------------------------------------------------------------------
x3|   1/5  |   0   -13/5     1     4/5    3/5   -2/5
x1|   7/5  |   1    -1/5     0     3/5    1/5    1/5
  +----------------------------------------------------------------------------

PHASE II
  +----------------------------------------------------------------------------
-z|  21/5  |   0   -13/5     0     9/5    3/5    3/5
  +----------------------------------------------------------------------------
x3|   1/5  |   0   -13/5     1     4/5    3/5   -2/5
x1|   7/5  |   1    -1/5     0     3/5    1/5    1/5
  +----------------------------------------------------------------------------

  THE PROBLEM IS UNBOUNDED
```

INTEGER LINEAR PROGRAMMING

```
  +----------------------------------------------------------------------------
-z|    4   |   0      0      0      1
  +----------------------------------------------------------------------------
x3|   1/3  |   0   -13/3     1     4/3
x1|   4/3  |   1     2/3     0     1/3
  +----------------------------------------------------------------------------
Generating row:           1
  +----------------------------------------------------------------------------
-z|    4   |   0      0      0      1      0
  +----------------------------------------------------------------------------
x3|   1/3  |   0   -13/3     1     4/3     0
x1|   4/3  |   1     2/3     0     1/3     0
x5|  -1/3  |   0    -2/3*    0    -1/3     1
  +----------------------------------------------------------------------------

  +----------------------------------------------------------------------------
-z|    4   |   0      0      0      1      0
  +-----------------------------------------------------------------------------
x3|   5/2  |   0      0      1     7/2  -13/2
x1|    1   |   1      0      0      0      1
x2|   1/2  |   0      1      0     1/2   -3/2
  +----------------------------------------------------------------------------
 Generating row:           1
  +----------------------------------------------------------------------------
-z|    4   |   0      0      0      1      0      0
  +----------------------------------------------------------------------------
x3|   5/2  |   0      0      1     7/2  -13/2     0
x1|    1   |   1      0      0      0      1      0
x2|   1/2  |   0      1      0     1/2   -3/2     0
x6|  -1/2  |   0      0      0    -1/2   -1/2*    1
  +----------------------------------------------------------------------------
  +----------------------------------------------------------------------------
-z|    4   |   0      0      0      1      0      0
  +----------------------------------------------------------------------------
x3|    9   |   0      0      1     10      0    -13
x1|    0   |   1      0      0     -1      0      2
x2|    2   |   0      1      0      2      0     -3
x5|    1   |   0      0      0      1      1     -2
  +----------------------------------------------------------------------------
```