Homework 3.2
10 points. End term: 12-th week (17-23 december)

1. Devise an implementation (C, C++, C#, Java)[1] for the Primal-Dual Interior Point Algorithm using the following pseudo-code:

$\varepsilon = 10^{-p}$; $k \leftarrow 0$; $q \leftarrow 6$;
$\theta \leftarrow 1 - 3.5/\sqrt{n}$; *// if $n > 13$, otherwise $\theta \leftarrow 0.5$.*
let $\mathbf{x}^k, \mathbf{s}^k > \mathbf{0}, \mathbf{y}^k$ **be initial estimates;** *//could be infeasible initial guess for the primal/dual solutions.*
let $\mu_k > 0$;
do {
$\quad \mathbf{S}_k \leftarrow diag\left(s_1^k, s_2^k, \ldots, s_n^k\right)$; *// $\mathbf{S}_k \in \mathbb{R}^{n\times n}$.*
$\quad \mathbf{D}_k \leftarrow diag\left(x_1^k/s_1^k, x_2^k/s_2^k, \ldots x_n^k/s_n^k\right)$; *// $\mathbf{D}_k \in \mathbb{R}^{n\times n}$.*
$\quad \rho_P^k = \mathbf{b} - \mathbf{A}\mathbf{x}^k$; *// $\rho_P^k \in \mathbb{R}^m$.*
$\quad \rho_D^k = \mathbf{c} - \mathbf{A}^T\mathbf{y}^k - \mathbf{s}^k$; *// $\rho_D^k \in \mathbb{R}^n$.*
$\quad \mathbf{v}^k = \left(\mu_k - x_1^k s_1^k, \mu_k - x_2^k s_2^k, \ldots, \mu_k - x_n^k s_n^k\right)^T$; *// $\mathbf{v}^k \in \mathbb{R}^n$.*
$\quad \Delta\mathbf{y}^k = -(\mathbf{A}\mathbf{D}_k\mathbf{A}^T)^{-1}\left[\mathbf{A}\mathbf{S}_k^{-1}\mathbf{v}^k - \mathbf{A}\mathbf{D}_k\rho_D^k - \rho_P^k\right]$; *// $\Delta\mathbf{y}^k \in \mathbb{R}^m$.*
$\quad \Delta\mathbf{s}^k = -\mathbf{A}^T\Delta\mathbf{y}_k + \rho_D^k$; *// $\Delta\mathbf{s}^k \in \mathbb{R}^n$.*
$\quad \Delta\mathbf{x}^k = \mathbf{S}_k^{-1}\mathbf{v}^k - \mathbf{D}_k\Delta\mathbf{s}_k$; *// $\Delta\mathbf{x}^k \in \mathbb{R}^n$.*
$\quad \alpha_{\mathbf{x}} \leftarrow \min\left\{-\frac{x_i^k}{\Delta x_i^k} \;:\; \Delta x_i^k < 0\right\}$; *// if $\Delta x_i^k \geqslant 0$, for all i, then $\alpha_{\mathbf{x}} \leftarrow 1$.*
$\quad \alpha_{\mathbf{s}} \leftarrow \min\left\{-\frac{s_i^k}{\Delta s_i^k} \;:\; \Delta s_i^k < 0\right\}$; *// if $\Delta s_i^k \geqslant 0$, for all i, then $\alpha_{\mathbf{s}} \leftarrow 1$.*
$\quad \alpha_{max} \leftarrow \min\{\alpha_{\mathbf{x}}, \alpha_{\mathbf{s}}\}$; $\alpha \leftarrow 0.999999\alpha_{max}$;
$\quad \mathbf{x}^{k+1} \leftarrow \mathbf{x}^k + \alpha\Delta\mathbf{x}^k$;
$\quad \mathbf{y}^{k+1} \leftarrow \mathbf{y}^k + \alpha\Delta\mathbf{y}^k$;
$\quad \mathbf{s}^{k+1} \leftarrow \mathbf{s}^k + \alpha\Delta\mathbf{s}^k$;
$\quad \mu_{k+1} \leftarrow \theta\mu_k$;
$\quad k{+}{+}$;
} while$\left(\left(\left(\mathbf{x}^k\right)^T\mathbf{s}^k > \varepsilon\right) \;\&\&\; (k < k_{max}) \;\&\&\left(||(\mathbf{x}^k, \mathbf{y}^k, \mathbf{s}^k) - (\mathbf{x}^{k-1}, \mathbf{y}^{k-1}, \mathbf{s}^{k-1})||_2 < 10^q\right)\right)$
if $\left(\left(\mathbf{x}^k\right)^T\mathbf{s}^k \leqslant \varepsilon\right)$
$\quad$ **return** $\mathbf{x}^k, \mathbf{y}^k$;

2. Run the above implemented algorithm on the following problems and verify the results using the Gurobi solver and your (already implemented) Simplex algorithm:

$$\textbf{(a)}\left\{\begin{array}{lrl} \min & z = x_1 + 3x_2 + x_3 & \\ \text{s. t.} & 2x_1 + x_2 + 3x_3 & = 35 \\ & -x_1 + x_2 + 2x_4 & = 12 \\ & x_i \geqslant 0 & \end{array}\right. , \quad \textbf{(b)}\left\{\begin{array}{lrl} \min & z = -2x_1 - x_2 & \\ \text{s. t.} & 2x_1 + x_2 + 2x_3 & = 4 \\ & 2x_1 + 3x_2 + x_4 & = 3 \\ & 4x_1 + x_2 + 3x_5 & = 5 \\ & x_1 + 5x_2 + x_6 & = 2 \\ & x_i \geqslant 0 & \end{array}\right. .$$

[1] You can use an appropriate numerical library to determine the inverse of a matrix, the euclidean norm etc.