

```

# load ".../Opt2011/MatrixUtils.py"
# load ".../Opt2011/step_3corrige.py"
# La règle de pivotage "least entering and exiting" est appliquée

sage: A = matrix(QQ,5,3,[1,1,-1,3,2,-4,0,0,3,4,-3,0,1,2,0])
sage: c = vector([2,2,1])
sage: b = vector([2,3,3,1,-2])
sage: G = matrix(QQ,6,3,[1,1,-1,3,2,-4,0,0,3,4,-3,0,1,2,0,2,2,1])
sage: h = vector([0,0,0,0,0,1])
sage: B_initial = [1,2,5]
sage: G.matrix_from_rows(B_initial).determinant()
-6
sage: x = G.matrix_from_rows(B_initial).solve_right(from_list(h,B_initial))
sage: G*x-h
(1/2, 0, 0, -17/2, 2, 0)
sage: B_initial = step_3(G,c,B_initial,0)
sage: x = G.matrix_from_rows(B_initial).solve_right(from_list(h,B_initial))
sage: G*x-h
(0, 0, 0, 0, 0, -1)
sage: B_initial
[1, 2, 0]
sage: x = A.matrix_from_rows(B_initial).solve_right(from_list(b,B_initial))
sage: A*x-b
(0, 0, 0, -3, 7)
sage: B = step_3(A,c,B_initial,4)
sage: x = A.matrix_from_rows(B).solve_right(from_list(b,B))
sage: A*x-b
(-7/4, 0, 0, 107/4, 0)
sage: B = step_3(A,c,B,3)
sage: x = A.matrix_from_rows(B).solve_right(from_list(b,B))
sage: A*x-b
(-46/11, -107/11, 0, 0, 0)
sage: z = A.matrix_from_rows(B).solve_left(c)
sage: z
(2/11, 1/3, 14/11)
sage: z*A.matrix_from_rows(B)
(2, 2, 1)

# solution optimale
sage: x
(-4/11, -9/11, 1)

# valeur de l'objectif
sage: z*A.matrix_from_rows(B)*x
-15/11

```