Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Кафедра информатики

Отчёт

Лабораторная работа №2

По учебной дисциплине *Методы оптимизации и управления*

Вариант 16

Выполнил:

студент группы №853504

Кузьма В.В.

Проверил:

доцент кафедры информатики

Дугинов О.И.

Минск 2021

ЗАДАНИЕ

Реализовать основную фазу симплекс метода

2.1.16

$$
\begin{aligned}
&25x_1 + 34x_2 + x_3 \to \max,\\
&5x_1 + 4x_2 + x_3 = 55,\\
&3x_1 + 7x_2 + x_4 = 56,\\
&x_1 + x_5 = 10,\\
&x_2 + x_6 = 7,\\
&x_1, x_2, x_3, x_4, x_5, x_6 \geq 0.
\end{aligned}
$$

$$x = (0, 0, 55, 56, 10, 7)'$$

Примеры работы

```
A=np.array([[5, 4, 1, 0,0,0], [3,7,0,1,0,0], [1,0,0,0,1,0],[0,1,0,0,0,1]])
b = np.array([0,0,0,0])
c = np.array([25,34,1,0,0,0])
x=np.array([0,0,55,56,10,7])
jb=np.array([3,4,5,6])
main_simplex(A,c,x,jb)
```

```
we have optimal plan
array([7., 5., 0., 0., 3., 2.])
```

Поэтапная реализация

Лаборатoрная работа №2 Курьма 13.13 853504

Вариант 16

$25x_1 + 34x_2 + x_3 \to max$

$5x_1 + 4x_2 + x_3 = 55$

$3x_1 + 7x_2 + x_4 = 56$

$x_1 + x_5 = 10$

$x_2 + x_6 = 7$

$x_1, x_2, x_3, x_4, x_5, x_6 \geq 0$

$x = (0, 0, 55, 56, 10, 7)$

$$A = \begin{pmatrix} 5 & 4 & 1 & 0 & 0 & 0 \\ 3 & 7 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$x = (0, 0, 55, 56, 10, 7)$

$c = (25, 34, 1, 0, 0, 0)$

$b = \begin{pmatrix} 55 \\ 56 \\ 10 \\ 7 \end{pmatrix}$ $J_B = (3, 4, 5, 6)$

Итерация 1

$$A_B = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad A_B^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$c_B' = (1, 0, 0, 0)$ $u' = c_B' \cdot A^{-1} = (1, 0, 0, 0) \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} =$

$= (1, 0, 0, 0)$

$$\Delta' = u' \cdot A - c' = (1, 0, 0, 0) \cdot \begin{pmatrix} 5 & 4 & 1 & 0 & 0 & 0 \\ 3 & 7 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix} - (25, 34, 1, 0, 0, 0) =$$

$= (-20, -30, 0, 0, 0, 0)$

$j_0 = 1$

$$z = A_B^{-1} \cdot A_{j_0} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 5 \\ 3 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 5 \\ 3 \\ 1 \\ 0 \end{pmatrix}$$

$$\theta = \begin{cases} \frac{x_{j_i}}{z_i}, & z_i \geq 0 \\ \infty, & z_i < 0 \end{cases} = \left(11, 18\frac{2}{3}, 10, \infty\right) \quad \theta_0 =$$

$j_{min\,\theta_0} = j_0$ $\quad J_б = (3, 4, 1, 6)$.

$x_{нов} = \left[\quad \middle| \begin{array}{l} x_{n j_0} = \theta_0 \\ x_{ji} = x_{ji} - \theta_0 \cdot z_i = (10, 0, 5, 26, 0, 7) \end{array} \right.$

Итерация 2

$A_б = \begin{pmatrix} 1 & 0 & 5 & 0 \\ 0 & 1 & 3 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$ $\quad A_б^{-1} = \begin{pmatrix} 1 & 0 & -5 & 0 \\ 0 & 1 & -3 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

$c_б = (1, 0, 25, 0)$ $\quad u = c_б \cdot A_б^{-1} = (1, 0, 25, 0) \cdot \begin{pmatrix} 1 & 0 & -5 & 0 \\ 0 & 1 & -3 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} =$

$= (1, 0, 20, 0)$

$\Delta = u \cdot A - c = (1, 0, 20, 0) \begin{pmatrix} 5 & 4 & 1 & 0 & 0 & 0 \\ 3 & 7 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix} - (25, 34, 1, 0, 0, 0) =$

$= (0, -30, 0, 0, 20, 0)$

$j_0 = 2$

$z = A_б^{-1} \cdot A_{j_0} = \begin{pmatrix} 1 & 0 & -5 & 0 \\ 0 & 1 & -3 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 4 \\ 7 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 4 \\ 7 \\ 0 \\ 1 \end{pmatrix}$

$\theta = \begin{cases} \dfrac{x_{iq}}{z_i}, & z_i > 0 \\ \infty, & z_i \le 0 \end{cases} \quad = (1.25, 3.715, \infty, 7)$

$\theta_0 = 1.25$ $\quad j_{nd} = j_i$ $\quad J_б = (2, 4, 1, 6)$

$x_{j_0} = \theta_0$

$x_{ji} = x_{ji} - \theta_0 \cdot z_i = (10, 1.25, 0, 17.25, 0, 5.75)$

Итерация 3

$A_б = \begin{pmatrix} 4 & 0 & 5 & 0 \\ 7 & 1 & 3 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}$ $\quad A_б^{-1} = \begin{pmatrix} 0{,}25 & 0 & -1{,}25 & 0 \\ -1{,}75 & 1 & 5{,}75 & 0 \\ 0 & 0 & 1 & 0 \\ -0{,}25 & 0 & 1{,}25 & 1 \end{pmatrix}$

$c = (34, 0, 25, 0)$

$u = (8{,}5, 0, -17.5, 0)$

$\Delta = (0, 0, 7.5, 0, -17.5, 0)$

$j_0 = 5$

$z = (-1.25, 5.75, 1, 1.25)$

$\theta = (\infty, 3, 10, 4.6)$

$\theta_0 = 3$

$J_б = (2, 5, 1, 6)$

$\boxed{X = (7.5, 0, 0, 3, 2)}$

Итерация 4

$\Delta = (0, 0, 2.17, 3, 04)$

Ответ: $(7{,}5, 0, 0, 3, 2)$

КОД ПРОГРАММЫ

```
def solve(A_inv, x, k):
  n = len(A_inv)
  k -= 1 # для лучшей индексации
  # ШАГ 1
  l = np.dot(A_inv, x) # вычисляем вектор l
  li = l[k][0] # находим i-ую компоненту вектора l
  if (li == 0): # проверка компоненты на равенство нолю
    return "Матрица не может быть обратимой"
  # ШАГ 2
  l1 = np.copy(l) # получаем вектор l с волной
  l1[k][0] = -1 # заменяем i-ую компоненту вектора l с волной -1
  # ШАГ 3
  l2 = np.dot(-1/li, l1) # получаем вектор l с шапочкой
  # ШАГ 4
  E = np.eye(n) # создаем единичную матрицу
  Q = np.copy(E)
  Q[:,k] = l2.transpose() # создаем матрицу Q
  # ШАГ 5
  z = np.eye(n)
  for i in range(n):
    for j in range(n):
      z[i][j] = Q[i][i]*A_inv[i][j]
      if (i != k):
        z[i][j] += Q[i][k]*A_inv[k][j]
  return z

def main_simplex(A, c, x, jb):
  iteration = 0
  while True:
    iteration += 1
    if iteration == 1:
      ab = np.array([np.copy(A[:,i-1]) for i in jb]).transpose()
      A_i = np.linalg.inv(ab)
    else:
      ab[:,position_min_tetta] = A[:, jb[position_min_tetta] - 1]
      A_i = solve(A_i, ab[:,position_min_tetta], position_min_tetta + 1)
    cb = np.array([c[i - 1] for i in jb])
    u = np.dot(cb, A_i)
    delta = np.dot(u, A) -c
    j0=0
    if min(delta) >= 0:
      print("we have optimal plan")
      return x
    while delta[j0]>=0:
      j0 += 1
    z = np.dot(A_i, A[:,j0])
    tetta = [float(x[jb[i] -
 1])/float(z[i]) if z[i]> 0 else np.inf for i in range(len(jb))]
    tetta0 = np.min(tetta)
    if tetta0 is np.inf:
      print("целевая функция неограничена на множестве допустимых планов"
)
    position_min_tetta = np.where(tetta == tetta0)[0][0]
```

```
jb_new = np.copy(jb)
jb_new[position_min_tetta] = j0 + 1
x_new = np.copy(x)
j = 0
x_new = x_new.astype(np.float64)
for i in jb:
  x_new[i - 1] = float(x[i - 1]) - tetta0*float(z[j])
  j+=1
x_new[j0] = tetta0
x = x_new
jb = jb_new
```