Рубежный контроль №1

1.1 Содержательная и математическая постановки задач о назначениях. Венгерский метод решения задачи о назначениях.

В распоряжении работодателя имеется n работ и n исполнителей. Стоимость выполнения i-й работы j-м исполнителем составляет $a_{ij} \geq 0$ единиц. Необходимо распределить работы между исполнителями таким образом, чтобы каждый исполнитель выполнял ровно 1 работу, а общая стоимость выполненных работ была бы минимальной.

Запишем стоимости в матрицу $C = (a_{ij})_{i,j=\overline{1:n}}$

Введём матрицу $X = (x_{i,j})_{i,j=\overline{1:n}}$, для которой определено условие

$$x_{i,j} = \begin{cases} 1, \text{ если i-ю работу выолняет j-й исполнитель} \\ 0, \text{ иначе} \end{cases}$$

Тогда общая стоимость выполнения работ $f = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x_{ij}$

Каждая строка матрицы X должна содержать ровно 1 единицу, т.к. каждую работу выполняет ровно 1 исполнитель.

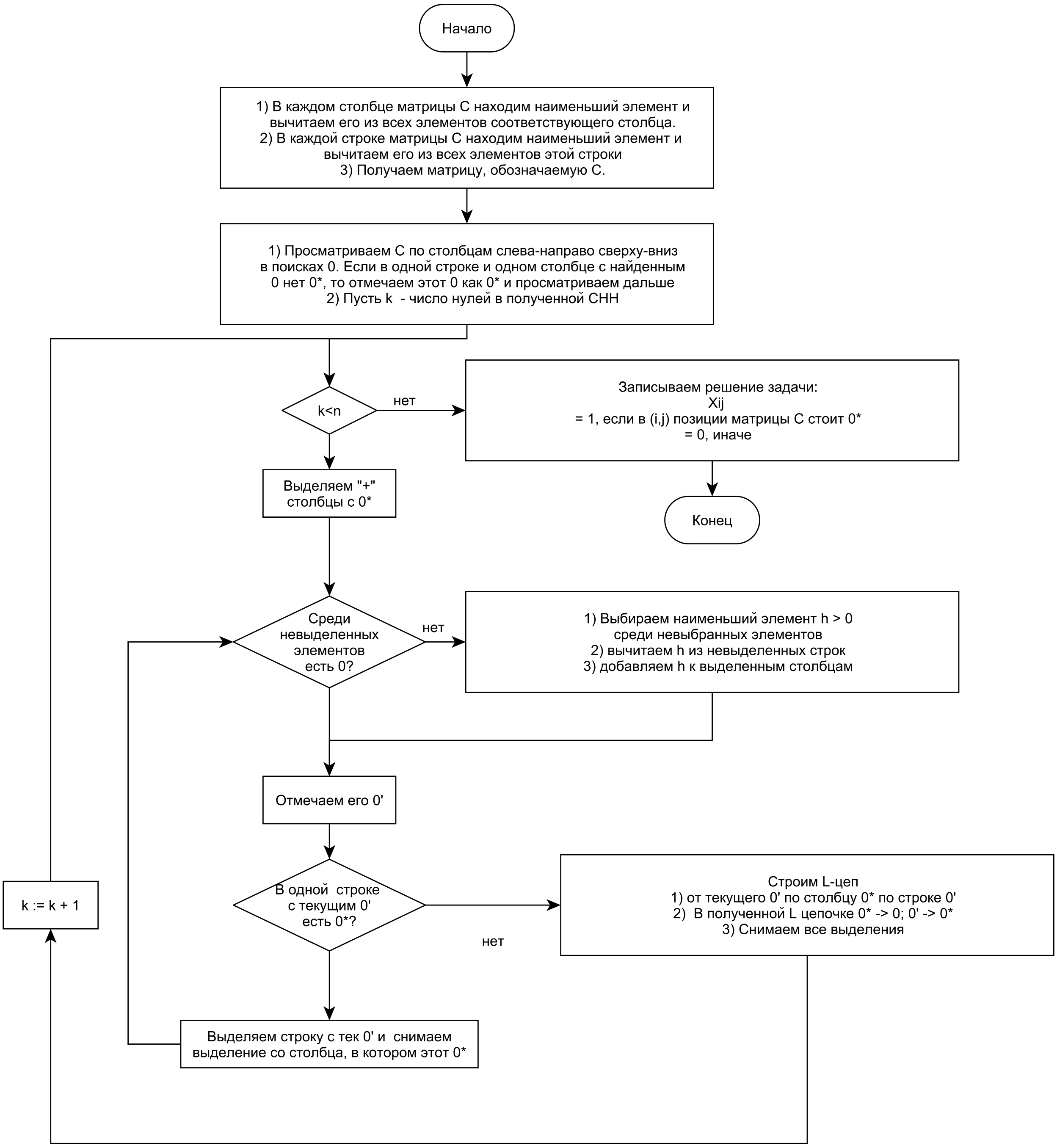
$\sum_{j=1}^{n} x_{ij} = 1$, где $i = \overline{1:n}$

Каждый столбец матрицы X должен содержать ровно 1 единицу:

$\sum_{i=1}^{n} x_{ij} = 1$, где $j = \overline{1:n}$

Таким образом приходим к математической постановке:

$$\begin{cases} f = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x_{ij} \to min \\ \sum_{j=1}^{n} x_{ij} = 1, \text{ где } i = \overline{1:n} \\ \sum_{i=1}^{n} x_{ij} = 1, \text{ где } j = \overline{1:n} \\ x_{ij} \in \{0, 1\} \end{cases}$$


Начало
1) В каждом столбце матрицы C находим наименьший элемент и вычитаем его из всех элементов соответствующего столбца.
2) В каждой строке матрицы C находим наименьший элемент и вычитаем его из всех элементов этой строки
3) Получаем матрицу, обозначаемую C.
1) Просматриваем C по столбцам слева-направо сверху-вниз в поисках 0. Если в одной строке и одном столбце с найденным 0 нет 0*, то отмечаем этот 0 как 0* и просматриваем дальше
2) Пусть k - число нулей в полученной СНН
k<n
нет
Записываем решение задачи:
Xij
= 1, если в (i,j) позиции матрицы C стоит 0*
= 0, иначе
Конец
Выделяем "+" столбцы с 0*
Среди невыделенных элементов есть 0?
нет
1) Выбираем наименьший элемент h > 0 среди невыбранных элементов
2) вычитаем h из невыделенных строк
3) добавляем h к выделенным столбцам
Отмечаем его 0'
В одной строке с текущим 0' есть 0*?
нет
Строим L-цеп
1) от текущего 0' по столбцу 0* по строке 0'
2) В полученной L цепочке 0* -> 0; 0' -> 0*
3) Снимаем все выделения
k := k + 1
Выделяем строку с тек 0' и снимаем выделение со столбца, в котором этот 0*

1.2 Общая постановка задачи линейного программирования. Стандартная форма задачи линейного программирования. Основные допущения, принимаемые при исследовании ЗЛП в стандартной форме. Доказать, что любая ЗЛП может быть сведена к стандартной форме.

Задача линейного программирования в общей форме имеет следующий вид

$$\begin{cases} f(x) = \sum_{k=1}^{n} c_k x_k \to extr \\ \sum_{j=1} n a_{ij} x_j \begin{Bmatrix} = \\ \geq \\ \leq \end{Bmatrix} b_i, \text{ где } i = 1..m \end{cases}$$

Стандартная форма

$$\begin{cases} f(x) = \sum_{k=1}^{n} c_k x_k \to max \\ \sum_{j=1} n a_{ij} x_j = b_i \geq 0, \text{ где } i = 1..m \\ x_j \geq 0, j = 1..n \end{cases}$$

Признаки стандартной формы:

1) целевая функция максимизируется

2) все ограничения имеют вид равенств с неотрицательными правыми частями.

3) все переменные неотрицательны.

Основные допущения, принимаемые при построении соответствующих математических моделей таковы:

1) Пропорциональность — для каждого вида производственной деятельности прибыль и затраты ресурсов строго пропорциональны объёму этой деятельности. При этом, подразумевается, что все производственно-экономические показатели могут принимать как целые, так и нецелые значения.

2) Аддитивность — если определён объём каждого вида производственной деятельности, то суммарная прибыль, равно как и суммарные затраты ресурсов, складываются из прибыли(затрат ресурсов) каждого отдельного вида производственной деятельности

3) Неотрицательности — ни одному из производственно-экономических показателей не может быть приписано отрицательное значение(запасы ресурсов, норма расходов ресурсов, объёмы производственной деятельности...)

Покажем, что любая задача линейного программирования может быть приведена к стандартной форме.

1) Если $f \to min$, то рассмотрим функцию $g = -f$, тогда эквивалентная задача будет иметь вид

$$\begin{cases} g \to max \\ \text{другие ограничения} \end{cases}.$$

2) Если некоторые ограничения являются неравенством $\sum_{j=1}^{n} a_{ij} x_j \leq b_i$, то введём дополнительную переменную $x_{n+1} \geq 0$, $\sum_{j=1}^{n} a_{ij} x_j + x_{n+1} = b_i$

3) Если некоторое ограничение является неравенством, но с другим знаком $\sum_{j=1}^{n} a_{ij} x_j \geq b_i$ то введём дополнительную переменную $x_{n+1} \geq 0$, $\sum_{j=1}^{n} a_{ij} x_j - x_{n+1} = b_i$

4) Если некоторое ограничение является равенством с отрицательной правой частью, т.е. имеет вид $\sum_{j=1}^{n} a_{ij} x_j = b_i, \text{ где } b_i < 0$, то умножим обе части этого равенства на -1

$\sum_{j=1}^{n} (-a_{ij}) x_j = -b_i, \text{ где } -b_i > 0$

5) Если некоторая переменная x_j не ограничена в знаке, т.е. не полчинена условию $x_j \geq 0$, то её

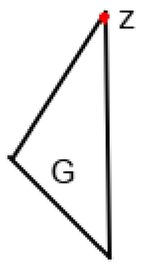
представляют в виде $\begin{cases} x_j = x'_j - x''_j \\ x'_j \geq 0 \\ x''_j \geq 0 \end{cases}$

■

1.3 Определение выпуклого множества и крайней точки выпуклого множества. Понятие выпуклой комбинации точек $q_1, ..., q_k \in \mathbb{R}^n$

Множество $G \subseteq \mathbb{R}^n$ называется выпуклым, если вместе с любыми двумя своими точками x и y G целиком содержит отрезок xy.

Точка z называется крайней точкой выпуклого множества G, если z не является внутренней точкой ни одного отрезка целиком лежащего в G.



Выпуклой комбинацией точек $q_1, ..., q_m \in \mathbb{R}^n$ называется множество точек вида $x = \sum_{i=1}^{m} \lambda_i q_i$, где $\sum_{i=1}^{n} \lambda_i = 1$ и $\lambda_i \geq 0$

1.4 Основные утверждения линейного программирования (формулировка). Доказать, что множество допустимых решений ЛЗП является выпуклым.

Рассмотрим задачу линейного программирования в стандатной форме.

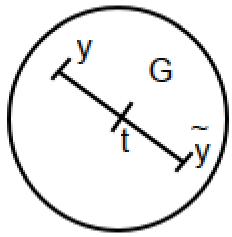
$$\begin{cases} f = c^T x \to max \\ Ax = b \\ x \geq 0 \end{cases}$$

Th 1 Пусть $G \neq 0$, тогда G содержит БДР

Th 2 Множество G допустимых решений ЗЛП является выпуклым.

Доказательство:

Пусть y и $\tilde{y}$ допустимое решение. Рассмотрим отрезок $t = \lambda y + (1-\lambda)\tilde{y}, \lambda \in [0,1]$.



Покажем, что $t \in G$

а) $At = A(\lambda y) + A(1-\lambda)\tilde{y} = \lambda Ay + A\tilde{y} - \lambda A\tilde{y} = \begin{Bmatrix} Ay = b \\ A\tilde{y} = b \end{Bmatrix} = \lambda b + b - \lambda b = b$

б) Т.к. $\lambda \in [0,1]$ и $y, \tilde{y} \geq 0$ то $t \geq 0 \Rightarrow t \in G$

■

Th 3 Пусть y БДР ЗЛП. Тогда y является крайней точкой мн-ва допустимых решений G.

Th 4 Пусть y — крайняя точка множества G, тогда y — БДР.

Th 5

1) Пусть целевая функция ЗЛП в стандартной форме принимает оптимальное значение хотя бы в одной точке множества допустимых решений G. Тогда f достигает этого значения в по крайней мере в одной крайней точке y множества допустимых решений G.

2) Если f достигает максимального значения в нескольких крайних точках мн-ва G, то она принимает это значение в любой выпуклой комбинации этих точек.

1.5 Понятия базисного решения и базисного допустимого решения задачи линейного программирования. Вычисление базисного решения и отвечающего ему значения целевой функции в случае, когда базисными выбраны m первых столбцов матрицы A.

(2) Базисное решение. Рассмотрим задачу линейненого программирования в стандартной форме.

$$\begin{cases} f = c^T x \to max \\ Ax = b \geq 0, \\ x \geq 0 \end{cases}$$

Без ограничения общности, будем считать, что базисный минор матрицы A расположен в m её первых столбцах(если это не так, то переставим и выполним соответствующую перенумерацию переменных).

$A = [[a_1]...[a_m][a_{m+1}]...[a_n]]$

Обозначим с a_1 до a_m — $A_Б$, остальные — $A_{НБ}$

$c = \begin{bmatrix} \begin{bmatrix} c_1 \\ ... \\ c_m \end{bmatrix} --\mathrm{C}_Б \\ \begin{bmatrix} ... \\ c_{m+1} \\ ... \\ c_n \end{bmatrix} --\mathrm{C}_{НБ} \end{bmatrix}$ С учётом этих обозначений нашу задачу линейного программирования можно записать в следующем виде:

$$\begin{cases} f = c_Б x_Б + {}_{НБ} x_{НБ} \to max \\ A_Б x_Б + A_{НБ} x_{НБ} = b \\ x_Б \geq 0, x_{НБ} \geq 0 \end{cases}$$

Рассмотрим СЛАУ из системы ограничений

$A_Б \mathrm{x}_Б + \mathrm{A}_{НБ} \mathrm{x}_{НБ} = b$

$A_Б \mathrm{x}_Б = b - \mathrm{A}_{НБ} \mathrm{x}_{НБ}$

$A_Б$ содержит m базисных столбцов матрицы A, $det(A_Б) \neq 0 \Rightarrow \exists \mathrm{A}_Б^{-1}$

Тогда $x_Б = \mathrm{A}_Б^{-1} b - A_Б^{-1} \mathrm{A}_{НБ} x_{НБ}$

Общее решение СЛАУ

$c = \begin{bmatrix} x_Б \\ \mathrm{x}_{НБ} \end{bmatrix} = \begin{bmatrix} \mathrm{A}_Б^{-1} b - A_Б^{-1} \mathrm{A}_{НБ} x_{НБ} \\ x_{НБ} \end{bmatrix}$

Неизвестные $\mathrm{x}_Б$ называют главными, или базисными, а переменные из $x_{НБ}$ называют небазисными, или свободными.

Переменные из $x_{НБ}$ могут принимать любые значения, каждому конкретному набору значений этих переменных отвечает некоторый набор значений из $\mathrm{x}_Б$

Если перебрать все возможные наборы значений небазисных переменных, для каждого из них найти $\mathrm{x}_Б$ и составить вектор x, то получим множество всех решений СЛАУ Ax=b

Опр.Базисным решением называется то частное решение Ax = b, которое отвечает $x_{\text{НБ}} = 0$, т.е.
$$x = \begin{bmatrix} A_{\text{Б}}^{-1}b \\ 0 \end{bmatrix}$$
Базисное решение СЛАУ Ax = b будет допустимым решением ЗЛП, если $x \geq 0$, т.е. $x_{\text{Б}} = A_{\text{Б}}^{-1}b \geq 0$. В этом случае базисное решение называется **базисным допустимым решением(БДР)**.

Замечание: 1) базисное решение однозначно определяется выбором базисных столбцов матрицы A. В общем случае базисных решений не превышает количества сочетаний из n по m.

2)любое базисное решение содержит по крайней мере n-m нулей.Если вдруг окажется, что базисное решение содержит больше n-m нулей (то есть некоторые базисные переменные = 0), то такое бащисное решение называется **вырожденным**.

1.6 Понятия базисного решения и базисного допустимого решения задачи линейного программирования. Каноническая форма ЗЛП в случае, когда базисными являются m первых столбцов матрицы A.

Первую часть смотри в 1.5

Обозначим соответствующие подматрицы

$A = [A_Б A_{НБ}]$

$C = \begin{pmatrix} С_Б \\ С_{НБ} \end{pmatrix}$

$x = \begin{bmatrix} x_Б \\ x_{НБ} \end{bmatrix}$

Тогда $x_Б = A_Б^{-1} b - \mathrm{A}_Б^{-1} \mathrm{A}_{НБ} \mathrm{x}_{НБ}$

$x = \begin{bmatrix} A_Б^{-1} b - \mathrm{A}_Б^{-1} \mathrm{A}_{НБ} \mathrm{x}_{НБ} \\ x_{НБ} \end{bmatrix}$

ЗЛП можно записать в следующем виде

$$\begin{cases} f = \mathrm{C}_Б^{\mathrm{T}} x_Б + C_{НБ}^{\mathrm{T}} x_{НБ} \\ A_Б x_Б + \mathrm{A}_{НБ} \mathrm{x}_{НБ} = b \\ \mathrm{x}_Б \geq 0 \\ \mathrm{x}_{НБ} \geq 0 \end{cases}$$

Подставим в задачу представление для $x_Б$

$f = \mathrm{C}_Б^{\mathrm{T}} (\mathrm{A}_Б^{-1} b - \mathrm{A}_Б^{-1} \mathrm{A}_{НБ} \mathrm{x}_{НБ}) + \mathrm{C}_{НБ} \mathrm{x}_{НБ} = \mathrm{C}_Б^{\mathrm{T}} \mathrm{A}_Б^{-1} b + (\mathrm{C}_{НБ} - \mathrm{A}_Б^{-1} \mathrm{A}_{НБ}) \mathrm{x}_{НБ}$

Следовательно, ЗЛП:

$$\begin{cases} f = \mathrm{C}_Б^{\mathrm{T}} \mathrm{A}_Б^{-1} b + (\mathrm{C}_{НБ} - \mathrm{A}_Б^{-1} \mathrm{A}_{НБ}) \mathrm{x}_{НБ} \to max \\ x_Б = \mathrm{A}_Б^{-1} b - \mathrm{A}_Б^{-1} \mathrm{A}_{НБ} \mathrm{x}_{НБ} \\ \mathrm{x}_Б \geq 0 \\ \mathrm{x}_{НБ} \geq 0 \end{cases}$$

Признаки канонической формы:

а) в выражении для целевой функции входят лишь небазисные переменные

б) базисные переменные выражены через небазисные.

1.7 Определение стандартной формы прямой задачи линейного программирования. Понятие двойственной задачи.

Стандартной формой прямой ЗЛП называется

$$\begin{cases} f(x) = c^T x \to max \\ Ax \le b \\ x \ge 0 \end{cases} \qquad (1)$$

Не обязательно b является неотрицательным в решении.

Понятие двойственности в линейном программировании позволяет установить взаимосвязи для анализа модели на чувствительность.

Каждой задаче линейного программирования отвечает некоторая другая задача линейного программирования, которая называется двойственной к ней. При этом исходную задачу называют прямой

Задачей, двойственной к (1) называют ЗЛП

$$\begin{cases} g(y) = b^T y \to min \\ A^T y \ge c \\ y \ge 0 \end{cases} \qquad (2)$$

1.8 Сформулируйте основные соотношения двойственности. Доказать, что задача, двойственная к двойственной, эквивалентна прямой задаче. Доказать утверждение о том, что целевая функция прямой задачи не превосходит целевую функцию двойственной задачи и его следствие.

Лемма 1 Задача, двойтсвенная к двойственной совпадает с прямой.

Доказательство

Прямая ЗЛП

$$\begin{cases} f(x) = c^T x \to max \\ Ax \leq b \\ x \geq 0 \end{cases}$$

Двойственная задача

$$\begin{cases} g(y) = b^T y \to min \\ A^T y \geq c \\ y \geq 0 \end{cases}$$

Двойственная в стандартной форме прямой задачи:

$$\begin{cases} -g(y) = -b^T y \to max \\ -A^T y \leq -c \\ y \geq 0 \end{cases}$$

Задача двойственная к двойственной:

$$\begin{cases} h = -c^t z \to min \\ (-A^T)^T z \geq -b \\ z \geq 0 \end{cases}$$

$$\Updownarrow$$

$$\begin{cases} h = c^t z \to max \\ Az \leq b \\ z \geq 0 \end{cases}$$

Совпадает с прямой задачей. ■

Лемма 2 Пусть 1) Прямая задача имеет допустмое решение x

2) Двойственняа задача имеет допустимое решение y

Тогда $f(x) \leq g(y)$

(Здесь f - целевая функция прямой задачи, g - целевая функция двойственной задачи)

Доказательство

1) Т.е. x допустим, то $Ax \geq b$

Умножим обе части неравенств на $y^T \geq 0$ слева: $\underset{\text{число}}{y^T Ax} \geq \underset{\text{число}}{y^T b}$

Транспонируем обе части $(y^T Ax)^T \leq (y^T b)^T$

$$x^T A^T y \leq b^T y == g(y)$$

Таким образом $x^T A^T y \leq g(y)$ (*)

2) Т.к. y допустим, то $A^T y \geq c$

Умножим обе части на $x^T \geq 0$ слева $x^T A^T y \geq x^T c$

Число => $x^T c = (x^T c)^T = c^T x = f(x)$

$\left| x^T A^T \geq f(x) \right|$ (**)

Из (*),(**) => $f(x) \leq g(y)$

■

<u>Следствие</u>: Если целевая функция прямой задачи не ограничена(сверху) в допустимой области, то множество допустимых решений двойственной задачи пусто.

<u>**Лемма 3**</u> Пусть 1) x_0 – допустмое решение прямой задачи

2) y_0 – допустмое решение двойственной задачи

3) $f(x_0) = g(y_0)$

Тогда 1) x_0 – оптимальное решение прямой задачи

2) y_0 – оптимальное решение двойственной задачи

<u>**Лемма 4**</u> Пусть 1) прямая задача имеет конечное оптимальное решение $f = f_{max}$

Тогда 1) Двойственная задача имеет конечное оптимальное решение $g = g_{min}$

2) $g_{min} = f_{max}$

<u>*Следствие*</u>: таким образом если двойственная задача имеет конечное оптимальное решение, то и прямая задача имеет конечное оптимальное решение.

<u>**Лемма 5**</u>: Пусть x_0, y_0 - оптимальные решения прямой и двойственной задачи соответственно. Тогда

$$y_i^0(\sum_{j=1}^{n} a_{ij}x_j^0 - b_i) = 0, i = 1..m$$

$$x_j^0(\sum_{i=1}^{m} a_{ij}y_i^0 - c_j) = 0, j = 1..n$$