1.Содержательная и математическая постановки задачи о назначениях. Венгерский метод решения задачи о назначениях.

В распоряжении работодателя находится n работ и n исполнителей. Стоимость выполнения i-й работы j-м исполнителем составляет $c_{ij} \geq 0$ единиц. Требуется распределить все работы между исполнителями так, чтобы каждый исполнитель выполнял ровно одну работу, а общая стоимость работ была минимальной.

Введём управляемые переменные: $x_{ij} = \begin{cases} 1, \text{если } i \text{ работу выполняет } j \text{ исполнитель,} \\ 0, \text{иначе} \end{cases}$

Тогда: Общая стоимость выполнения всех работ $f = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$

Условие того, что j-й исполнитель выполняет одну работу $\sum_{i=1}^{n} x_{ij} = 1, j = \overline{1,n}$

Условие, что i-ю работу выполняет один исполнитель $\sum_{j=1}^{n} x_{ij} = 1, i = \overline{1,n}$

Матрица стоимостей C = (cij), где i, j = 1; n. Матрица назначений X = (xij), где i, j = 1; n.

Математическая постановка задачи о назначениях:

$$\begin{cases} f = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \to \min, \\ \sum_{i=1}^{n} x_{ij} = 1, \quad j = \overline{1,n} \\ \sum_{j=1}^{n} x_{ij} = 1, \quad i = \overline{1,n} \\ x_{ij} \in \{0,1\}, \quad i,j = \overline{1,n} \end{cases}$$

Венгерский метод:

1. В каждом *столбце* матрицы C найти наименьший элемент и вычесть его из всех элементов *столбца*
2. В каждой *строке* полученной матрицы найти наименьший элемент и вычесть его из элементов *строки*
3. Просматриваем полученную матрицу по **столбцам** в поисках нулей, в одной **строке** с которыми нет 0*. Найденные нули **отмечать** 0*, k = число 0* в полученной матрице
4. **k = n?**

Если да:

 5. сформировать матрицу X_{opt}: $x_{ij} = 1$, если в соответствующей позиции полученной матрицы стоимостей стоит 0*; $x_{ij} = 0$ иначе

 6. вычислить $f_{opt} = f(X_{opt})$

Если нет:

 7. отмечаем «+» **столбцы с 0***

 8. среди **невыделенных** элементов есть **0**?

 Если да:

  9. отмечаем его **0'**

  10. в одной **строке** с 0' есть **0*** ?

  Если да:

   11. **снимаем** выделение со **столбца** с этим 0*

   выделяем + **строку** с 0'

   GOTO 8

  Если нет:

   12. строим непродолжаемую **L-цепочку**: от выбранного 0', по столбцу до 0*, по строке до другого 0', …

   13. в пределах этой L-цепочки **заменяем** 0* на 0', а 0' на 0*

   14. **снимаем** все выделения (' и +), оставляем только 0*

15. k += 1
GOTO 4

Если нет:

16. выбираем **наименьший** элемент h>0 среди всех **невыделенных «+»** элементов
вычитаем h из **невыделенных строк** и **добавляем** к **выделенным столбцам**
GOTO 8

Венгерский метод решения задачи о назначениях применяется для решения задачи минимизации, то есть для случая, когда матрица стоимостей C содержит в себе стоимости работ. Также матрица стоимостей C может использоваться для решения задачи максимизации, в этом случае её элементы отражают прибыль работодателя от назначения j-го работника на i-ю работу. В таком случае в начале алгоритма добавляется шаг, на котором необходимо найти максимальный элемент столбца и прибавить его к элементам, умноженным на -1.

2.Общая постановка задачи линейного программирования. Стандартная форма задачи линейного программирования. Основные допущения, принимаемые при исследовании задачи линейного программирования в стандартной форме. Показать, что любая задача линейного программирования может быть приведена к стандартной форме.

Общая математическая постановка ЗЛП:

$$\begin{cases} f = \sum_{j=1}^{n} c_j x_j \to extr \\ \sum_{j=1}^{n} a_{ij} x_j \begin{Bmatrix} \geq \\ \leq \\ = \end{Bmatrix} b_i, \qquad i = \overline{1,m} \end{cases}$$

Говорят, что ЗЛП записана в ***стандартной форме***, если она имеет вид:

$$\begin{cases} f = \sum_{j=1}^{n} c_j x_j \to max \\ \sum_{j=1}^{n} a_{ij} x_j = b_i, \qquad i = \overline{1,m} \end{cases}$$

Рассматривая стандартную форму ЗЛП, делают следующие ***допущения***:
1) Ранг матрицы A равен рангу блочной матрицы: $\mathbf{rg}\, A = \mathbf{rg}(A|b)$. В соответствие с критерием Кронекера-Копелли, СЛАУ $Ax = b$ совместна тогда и только тогда, когда ранг матрицы системы равен рангу расширенной матрицы этой системы. Если это условие не выполнено, то СЛАУ, входящая в систему ограничений $Ax = b$, несовместна – т.е. не имеет решений. В этом случае, множество допустимых решений ЗЛП пусто – оптимизировать нечего. **2) $\mathbf{rg}\, A = m$ – числу ограничений.**Если $\operatorname{rg} A < m$, то некоторые ограничения являются линейными комбинациями остальных – следовательно, их можно отбросить, не изменив множество допустимых решений. **3)Число переменных $n > m$.** Если $m = n$, то существует ровно одно единственное решение системы $Ax = b$. Следовательно, ЗЛП имеет не более 1 допустимого решения, а задача оптимизации вырождена.

Докажем, что любая ЗЛП может быть приведена к стандартной форме.
1) Если целевая функция **минимизируется**, то переходим к функции $f_1 = -f = \sum_{j=1}^{n}(-c_j)x_j \to max$
2) Если некоторое ограничение содержит **отрицательную правую часть**, перейдём к эквивалентному, умножив обе части на -1: $\sum_{j=1}^{n}(-a_{ij})x_j = -b_i$

3) Если некоторое ограничение имеет вид **неравенства ≤,** то добавим к левой части дополнительную переменную $x_{n+1} \geq 0$ и выравняем неравенство до равенства:

$$\begin{cases} \sum_{j=1}^{n} a_{ij}x_j + x_{n+1} = b_i \\ \qquad\qquad x_{n+1} \geq 0 \end{cases}$$

4) Аналогично, если ограничение имеет вид **неравенства ≥,** то вычтем дополнительную переменную:

$$\begin{cases} \sum_{j=1}^{n} a_{ij}x_j - x_{n+1} = b_i \\ \qquad\qquad x_{n+1} \geq 0 \end{cases}$$

5) Если некоторая переменная **отрицательна**, то введем вместо неё $x_j' = -x_j \geq 0$

6) Если переменная **не ограничена в знаке**, то представим её в виде разности дополнительных неотрицательных переменных:

$$\begin{cases} x_j = x_j' - x_j'' \\ x_j' \geq 0 \\ x_j'' \geq 0 \end{cases}$$

3.Определение выпуклого множества и крайней точки выпуклого множества. Понятие выпуклой комбинации точек q1, . . . , qk ∈ R n . Свойства выпуклой комбинации.

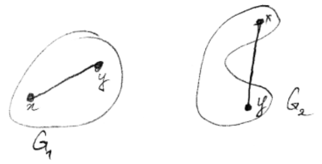
Отрезком, соединяющим две точки $x, y \in \mathbb{R}^n$, называется следующее множество точек:

$xy = \{t \in \mathbb{R}^n:\ t = \lambda x + (1-\lambda)y, \lambda \in [0; 1]\}$

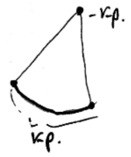
$\lambda = 0 \ \Rightarrow \ t(\lambda) = y$

$\lambda = 1 \ \Rightarrow \ t(\lambda) = x$

Множество $G \subseteq R^n$ называется ***выпуклым***, если вместе с любыми двумя своими точками x, y множество G целиком содержит соединяющий их отрезок. Т.е. $\forall\forall x, y \in G \ \ xy \subseteq G$.



Точка $x \in G$ выпуклого множества G называется ***крайней точкой*** этого множества G, если x не содержится ***строго*** *внутри* никакого отрезка, целиком лежащего в G.



Выпуклой комбинацией точек $q_1, \dots q_k \in R^n$ называется следующее множество:

К примеру, выпуклая комбинация двух точек – отрезок; трех точек – треугольник.

$< q1, \dots, qk >= x \in R\ n : x = \sum k\ j=1\ \lambda j * qj,\ \lambda j > 0, j = 1; k; \sum k\ j=1\ \lambda j = 1.$

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

Рассмотрим ЗЛП в стандартной форме $\begin{cases} f = cx \to max \\ Ax = b \geq 0 \\ x \geq 0 \end{cases}$ и обозначим $G = \{x \in \mathbb{R}^n: Ax = b, x \geq 0\}$ – множество допустимых решений этой задачи. ***Основные утверждения*** линейного программирования:

1)G – **выпуклое** множество; 2)Если G **не пусто**, то оно содержит **БДР** ЗЛП; 3)**БДР** ЗЛП являются **крайними точками** множества G; 4)**Крайние** точки множества G являются **БДР**;
5) 1. Пусть функция f принимает **максимальное** значение в **некоторой** точке G – тогда f принимает это значение **по крайней мере** в одной **крайней** точке G.
2. Если f принимает **максимальное** значение в точках $q_1, \dots, q_k$, то f принимает это значение в любой точке их **выпуклой комбинации** $\langle q_1, \dots, q_k \rangle$.

Докажем, что множество G допустимых решений ЗЛП является выпуклым.

Выберем две точки $x, y \in G$. Если G – выпуклое множество, то $\forall \lambda \in (0;1)$ $t = \lambda x + (1-\lambda) y, t \in G$. Покажем это:

$$\begin{aligned} At &= A(\lambda x + (1-\lambda)y) \\ &= \lambda Ax + (1-\lambda)Ay \\ &\quad \begin{vmatrix} Ax = b \\ Ay = b \end{vmatrix} \\ &= \lambda b + (1-\lambda) b \\ &= b \end{aligned}$$

Также покажем, что t удовлетворяет ограничению неотрицательности СЛАУ. $t = \begin{bmatrix} t_1 \\ \vdots \\ t_n \end{bmatrix}, t_j = \lambda x_j + (1-\lambda) y_j$. $x_j, y_j \geq 0,\ \lambda \in (0;1)$. Следовательно, оба слагаемых ≥ 0, а следовательно, $t_j \geq 0$.

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 системы ограничений задачи.

Рассмотрим СЛАУ $Ax = b$ из ЗЛП в стандартной форме $\begin{cases} f = c^T x \to max, \\ Ax = b \geq 0, \\ x \geq 0 \end{cases}$. Предположим, что m первых столбцов матрицы A линейно независимы – тогда их можно выбрать в качестве базисных столбцов. Обозначим:

$$A = [\underbrace{a_1, \dots, a_m}_{A_{\text{Б}}}, \underbrace{a_{m+1}, a_n}_{A_{\text{НБ}}}], \qquad x = \begin{bmatrix} x_1 \\ \vdots \\ x_m \\ x_{m+1} \\ \vdots \\ x_n \end{bmatrix} \begin{matrix} \left.\vphantom{\begin{matrix} x_1 \\ \vdots \\ x_m \end{matrix}}\right\} X_{\text{Б}} \\ \left.\vphantom{\begin{matrix} x_1 \\ \vdots \\ x_m \end{matrix}}\right\} X_{\text{НБ}} \end{matrix}$$

(a_j – столбцы). Запишем СЛАУ $Ax = b$ в векторной форме: $x_1 a_1 + \cdots x_m a_m + x_{m+1} a_{m+1} + \cdots + x_n a_n = b$. Т.е.: $A_{\text{б}} X_{\text{б}} + A_{\text{нб}} X_{\text{нб}} = b$
Выразим из получившегося равенства $X_{\text{б}}$. Столбцы матрицы $A_{\text{б}}$ линейно независимы, а сама матрица имеет размеры $m \times m$: следовательно, её определитель $\neq 0$ и существует обратная матрица $A_{\text{б}}^{-1}$. Тогда, $X_{\text{б}} = A_{\text{б}}^{-1}(b - A_{\text{нб}} X_{\text{нб}}) = A_{\text{б}}^{-1} b - A_{\text{б}}^{-1} A_{\text{нб}} X_{\text{нб}}$ $(*)$
Мы выразили базисные переменные через небазисные. Тогда решение всей системы можно записать в виде

$$x = \begin{bmatrix} A_{\text{б}}^{-1} b - A_{\text{б}}^{-1} A_{\text{нб}} X_{\text{нб}} \\ X_{\text{нб}} \end{bmatrix}$$

Небазисные переменные в векторе $X_{\text{нб}}$ могут принимать любые значения. Каждому конкретному набору значений этих переменных отвечает некоторый набор значений базисных переменных (исходя из формулы $(*)$) – а значит, некоторое решение СЛАУ $Ax = b$. Совокупность всех этих решений образует общее решение этой линейной системы.

<u>Базисным решением</u> данной линейной системы называется то ее частное решение, которое отвечает нулевым значениям небазисных переменных: $X^{\text{баз}} = x|_{\text{НБ}=0} = \begin{bmatrix} A_{\text{Б}}^{-1}b \\ 0 \end{bmatrix}$.
Если это базисное решение $X^{\text{баз}} \geq 0$, то оно называется ***<u>базисным допустимым решением</u>*** ЗЛП.

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

Рассмотрим ЗЛП в стандартной форме $\begin{cases} f = c^T x \to max, \\ Ax = b \geq 0, \\ x \geq 0 \end{cases}$. Будем считать, что известно некоторое БДР, которое отвечает m первым столбцам матрицы A. Обозначим

$$A = [\underbrace{a_1 \dots a_m}_{A_{\text{Б}}} \underbrace{a_{m+1} \dots a_n}_{A_{\text{НБ}}}], \qquad x = \begin{bmatrix} x_1 \\ \vdots \\ x_m \\ x_{m+1} \\ \vdots \\ x_n \end{bmatrix} \begin{matrix} \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} X_{\text{Б}} \\ \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} X_{\text{НБ}} \end{matrix}, \qquad c = \begin{bmatrix} c_1 \\ \vdots \\ c_m \\ c_{m+1} \\ \vdots \\ c_n \end{bmatrix} \begin{matrix} \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} C_{\text{Б}} \\ \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} C_{\text{НБ}} \end{matrix}$$

$$Ax = b \iff A_{\text{Б}}X_{\text{Б}} + A_{\text{НБ}}X_{\text{НБ}} = b$$

Выразим базисные переменные через небазисные: $X_{\text{Б}} = A_{\text{Б}}^{-1}b - A_{\text{Б}}^{-1}A_{\text{НБ}}X_{\text{НБ}}$. Тогда значение функции f

$$\begin{aligned} f &= C_{\text{Б}}^T X_{\text{Б}} + C_{\text{НБ}}^T X_{\text{НБ}} \\ &\qquad |\text{подставляя } X_{\text{Б}}| \\ &= C_{\text{Б}}^T A_{\text{Б}}^{-1}b - C_{\text{Б}}^T A_{\text{Б}}^{-1}A_{\text{НБ}}X_{\text{НБ}} + C_{\text{НБ}}^T X_{\text{НБ}} \\ &= \underbrace{C_{\text{Б}}^T A_{\text{Б}}^{-1}b}_{f_0} + \underbrace{(C_{\text{НБ}}^T - C_{\text{Б}}^T A_{\text{Б}}^{-1}A_{\text{НБ}})}_{[d_{m+1},\dots,d_n]} X_{\text{НБ}} \end{aligned}$$

Тогда мы можем представить ЗЛП в следующем виде:

$$\begin{cases} f = f_0 + d_{m+1}x_{m+1} + \dots + d_n x_n \\ \begin{bmatrix} x_1 \\ \vdots \\ x_m \end{bmatrix} = \begin{bmatrix} \propto_1 \\ \vdots \\ \propto_m \end{bmatrix} - x_{m+1}t_{m+1}^{\downarrow} - \dots - x_n t_n^{\downarrow}, \\ x_i \geq 0, \qquad i = \overline{1,n} \end{cases} \qquad \begin{matrix} \begin{bmatrix} \propto_1 \\ \vdots \\ \propto_m \end{bmatrix} = A_{\text{Б}}^{-1}b \\ [t_{m+1}^{\downarrow}, \dots, t_n^{\downarrow}] = A_{\text{Б}}^{-1}A_{\text{НБ}} \end{matrix}$$

Данную систему называют ***<u>канонической формой</u>*** ЗЛП, отвечающей выбранному базису. В ней базисные переменные выражены через небазисные, а выражение для целевой функции содержит только небазисные переменные.

7.Определение стандартной формы прямой задачи линейного программирования. Понятие двойственной задачи. Показать, что любая задача линейного программирования может быть приведена к стандартной форме прямой задачи.

Стандартная форма ЗЛП имеет вид $\begin{cases} f = c^T x \to max \\ Ax = b \geq 0 \\ x \geq 0 \end{cases}$. По определению, ***<u>стандартной формой прямой задачи</u>*** называется задача следующего вида (условие неотрицательности вектора b может не выполняться): $\begin{cases} f = c^T x \to max \\ Ax \leq b \\ x \geq 0 \end{cases}$,

<u>Двойственной</u> к данной задаче будет называться ЗЛП вида

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

Любая ЗЛП может быть приведена к стандартной форме прямой задачи путем домножения целевой функции и необходимых ограничений на -1 – таким образом можно перейти от минимизации к максимизации, а от неравенств $\geq$ к $\leq$. В случае, если какое-то из ограничений является равенством $\sum_{j=1}^{n} a_{ij}x_j = b_i$, можно перейти к двум неравенствам: $a_{i1}x_1 + \cdots + a_{in}x_n \leq b_i$ и $-a_{i1}x_1 - \cdots - a_{in}x_n \geq -b_i$. Если одна из переменных неограничена в знаке, её можно представить в виде в виде разности двух положительных переменных.

8.Понятие двойственной задачи. Сформулировать основные соотношения двойственности. Доказать, что задача, двойственная к двойственной, эквивалентна прямой задаче.

Стандартная форма ЗЛП имеет вид $\begin{cases} f = c^T x \to max \\ Ax = b \geq 0 \\ x \geq 0 \end{cases}$. По определению, ***<u>стандартной формой прямой задачи</u>*** называется задача следующего вида (условие неотрицательности вектора b может не выполняться): $\begin{cases} f = c^T x \to max \\ Ax \leq b \\ x \geq 0 \end{cases}$,

<u>Двойственной</u> к данной задаче будет называться ЗЛП вида

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

Основные соотношения двойственности:

1)Задача, **двойственная** к **двойственной**, эквивалентна прямой задаче; **2)**Если **прямая** задача имеет допустимое решение x^0 и **двойственная** имеет допустимое решение y^0, то $\boldsymbol{f(x^0)} \leq \boldsymbol{g(y^0)}$; **Следствие**: Если целевая функция **прямой** задачи **не ограничена сверху** на множестве допустимых решений, то **двойственная** задача **существует**, но **не имеет допустимых** решений.

3)Пусть x^0, y^0 – допустимые решения прямой и двойственной задачи, $\boldsymbol{f(x^0)} = \boldsymbol{g(y^0)}$. Тогда x^0, y^0 – **оптимальные** решения своих задач. **4)**Пусть прямая задача имеет конечное **оптимальное** решение x^0. Тогда двойственная задача **имеет** конечное оптимальное решение y^0, при этом $\boldsymbol{f_{max}} = \boldsymbol{f(x^0)} = \boldsymbol{g(y^0)} = \boldsymbol{g_{min}}$ **5)**Пусть $x^0 = (x_1^0, \ldots, x_n^0)^T$ – оптимальное решение прямой задачи, $y^0 = (y_1^0, \ldots, y_m^0)^T$ – двойственной. Тогда:

$$x_j^o \left(\sum_i^m a_{ij} y_i^o - c_j \right) = 0, \qquad y_i^o \left(\sum_j^n a_{ij} x_j^o - b_i \right) = 0$$

<u>Теорема1</u>: задача, двойственная к двойственной, эквивалентна прямой.

<u>Доказательство</u>:

Прямая задача	Двойственная
$\begin{cases} f = c^T x \to max \\ Ax \leq b \\ x \geq 0 \end{cases}$	$\begin{cases} g = b^T y \to min \\ A^T y \geq c \\ y \geq 0 \end{cases}$

Запишем двойственную задачу в стандартной форме прямой: $\begin{cases} -g = (-b)^T y \to max \\ (-A)^T y \le c \\ y \ge 0 \end{cases}$.

Тогда двойственная к этой задаче будет выглядеть как

$$\begin{cases} h = (-c)^T z \to min \\ (-A)^T z \ge -b \\ z \ge 0 \end{cases} \Leftrightarrow \begin{cases} -h = c^T z \to max \\ Az \le b \\ z \ge 0 \end{cases} - \text{прямая.}$$

<u>Теорема2</u>: пусть прямая задача имеет допустимое решение x^0, двойственная y^0. Тогда, $f(x^0) \le g(y^0)$

<u>Доказательство</u>:

Прямая задача Двойственная

$$\begin{cases} f = c^T x \to max \\ Ax \le b \\ x \ge 0 \end{cases} \quad \begin{cases} g = b^T y \to min \\ A^T y \ge c \\ y \ge 0 \end{cases}$$

Т.к. x_0 – допустимое решение, то $Ax_0 \le b$.

Умножим обе части неравенства слева на $(y^0)^T$, причем этот вектор неотрицателен.

$$(y^0)^T A x^0 \le (y^0)^T b$$

Если мы транспонируем обе части, то получим $((y^0)^T A x^0)^T \le ((y^0)^T b)^T$

Т.к. транспонирование произведения матриц есть произведение транспонированных в обратном порядке, то $\Rightarrow (x^0)^T A^T y^0 \le b^T y^0 \ (*)$

y_0 – допустимое решение двойственной задачи, следовательно $A^T y^0 \ge c$

В этом неравенстве умножим обе части слева на $(x^0)^T \ge 0$: $(x^0)^T A^T y^0 \ge (x^0)^T c \ \ (**)$

Наконец, из * и ** получаем: $(x^0)^T c \le (x^0)^T A^T y^0 \le b^T y^0 \ \Rightarrow \ \underbrace{(x^0)^T c}_{=f(x^0)} \le \underbrace{b^T y^0}_{=g(y^0)}$

Т.е., $f(x0) \le g(y0)$

<u>Следствие</u>: если целевая функция прямой задачи не ограничена сверху на множестве допустимых решений, то двойственная задача существует, но не имеет допустимых решений.