СОДЕРЖАНИЕ

ВВЕДЕНИЕ 5

1. ОБЩАЯ ЧАСТЬ 8

1.1 Математическая постановка задачи 8

1.2 Алгоритм решения задачи 11

1.3 Блок-схема (алгоритм решения) 25

2. Формы входной информации 27

3. Формы выходной информации 28

4. Инструкция для пользователя 29

5. Инструкция для программиста 30

ЗАКЛЮЧЕНИЕ 33

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 34

ПРИЛОЖЕНИЕ А 35

ВВЕДЕНИЕ

Математика необходима в повседневной жизни, следовательно определенные математические навыки нужны каждому человеку. Нам приходится в жизни считать(например, деньги), мы постоянно используем(часто не замечая этого) знания о величинах, характеризующих протяженности, площади, объемы, промежутки времени, скорости и многое другое. Все это пришло к нам на уроках арифметики и геометрии и пригодилось для ориентации в окружающем мире.

Математические знания и навыки нужны практически во всех профессиях, прежде всего, конечно, в тех, что связаны с естественными науками, техникой и экономикой. Математика является языком естествознания и техники и потому профессия естествоиспытателя и инженера требует серьезного овладения многими профессиональными сведениями, основанными на математике.

Хорошо сказал об этом Галилей:

«Философия (на нашем языке- физика) написана в величайшей книге, которая постоянно открыта вашему взору, но понять ее может лишь тот, кто сначала научится понимать ее язык и толковать знаки, которыми она написана. Написана же она на языке математики».

Сегодня несомненна необходимость применения математических знаний и математического мышления врачу, лингвисту, историку, и людям других специальностей. Но особенно знание математики необходимы людям точных профессий - финансистам, экономистам.

Профессиональный уровень экономиста во многом зависит от того, освоил ли он современный математический аппарат и умеет ли использовать его при анализе сложных экономических процессов и принятий решений. Поэтому в подготовке экономистов широкого профиля изучения математики занимает значительное место. Математическая подготовка экономиста имеет свои особенности, связанные со спецификой экономических задач, а также с широким разнообразием подходов к их решению.

Задачи практической и теоретической экономики очень разносторонни. К ним относятся, в первую очередь, методы сбора и обработки статической информации, а также оценка состояния и перспективы развития экономических процессов. Применяются различные способы использования полученной информации - от простого логического анализа до составления сложных экономико-математических моделей и разработки математического аппарата их исследования.

Неопределенность экономических процессов, значительный случайный разброс и большой объем получаемой информации обуславливают необходимость привлечения к исследованию экономических задач теории вероятностей и математической статистики.

Наряду с моделированием экономистам необходимо изучать теорию оптимизации, которая представлена математическими методами исследования операций, в том числе линейным программированием.

Отмеченные направления требуют знания основополагающего математического аппарата: основ линейной алгебры и математического анализа, теории вероятностей и математического программирования.

Таким образом, математика и математическое образование нужны для подготовки к будущей профессии.

Один из классов математических моделей- задачи линейного программирования. Одной из задач линейного программирования является транспортная задача- задача составления оптимального плана перевозок, позволяющего минимизировать суммарный километраж. Транспортная задача, как и задача линейного программирования была впервые поставлена советским экономистом А.Н.Толстым в 1930 году. Разработка общих методов решения задачи линейного программирования и их математическое исследование связано с именем советского ученого Л.В.Канторовича. В 1939 году методам решения задачи линейного программирования посвящено также большое число работ зарубежных ученых. Основной метод решения задачи линейного программирования –симплекс метод- был опубликован в 1949 году Дандигом. Симплекс метод дает решение любой задачи линейного программирования, но если переменных очень много, то решение весьма затруднительно и для более сложных задач симплекс метод стали модифицировать.

Транспортная задача делится на два вида: транспортная задача по критерию стоимости- определение плана перевозок, при котором стоимость груза была бы минимальна; транспортная задача по критерию времени- более важным является выигрыш по времени.

Транспортная задача по критерию стоимости является частным случаем задачи линейного программирования и может быть решена симплексным методом. Однако в силу особенностей задачи, она решается намного проще.

1. ОСНОВНАЯ ЧАСТЬ
	1. МАТЕМАТИЧЕСКАЯ ПОСТАНОВКА ЗАДАЧИ

Транспортная задача-

Однородный груз сосредоточен у *т* поставщиков в объемах .

Данный груз необходимо доставить *п* потребителям в объемах .

Известны (i=1,2,…,m; j=1,2,…,n)- стоимости перевозки единицы груза от каждого i-го поставщика каждому j-му потребителю. Требуется составить такой план перевозок, при котором запасы всех поставщиков вывозятся полностью, запросы всех потребителей удовлетворяются полностью и суммарные затраты на перевозку всех грузов минимальны.

Исходные данные транспортной задачи записываются в таблице вида

Таблица 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |   |   |  **…** |   |
|  |   |   |  **…** |   |
|  |   |   |  **…** |   |
| **…** |  **…** |  **…** |  **…** |  **…** |
|  |   |   |  **…** |   |

Переменными(неизвестными) транспортной задачи являются (i=1,…,m;i=1,2,…,n)- объемы перевозок от каждого i-го поставщика каждому j-му потребителю. Эти переменные могут быть записаны в матрице перевозок

Математическая модель транспортной задачи в общем случае имеет вид

 (1.1)

 i=1,2,…,m, (1.2)

 j=1,2,…,n, (1.3)

 i=1,2,…,m; j=1,2,…,n. (1.4)

Целевая функция задачи (1.1) выражает требования обеспечить минимум суммарных затрат на перевозку всех грузов. Первая группа из *т* уравнений (1.2) описывает тот факт, что запасы всех *т* поставщиков вывозятся полностью. Вторая группа из n уравнений (1.3) выражает требования полностью удовлетворить запросы всех n потребителей. Неравенства (1.4) являются условиями неотрицательности всех переменных задачи.

Таким образом, математическая формулировка транспортной задачи состоит в следующем: найти переменные задачи

 i=1,2,…,m; j=1,2,…,n,

удовлетворяющее системе ограничений (1.2), (1.3), условиям неотрицательности (1.4) и обеспечивающее минимум целевой функции (1.1).

В рассмотренной модели транспортной задачи предполагается, что суммарные запасы поставщиков равны суммарным запросам потребителей, т.е.

.

Такая задача называется задачей с правильным балансом, а ее модель- закрытой. Если же это неравенство не выполняется, то задача называется задачей с неправильным балансом, а ее модель- открытой.

Для того чтобы транспортная задача линейного программирования имела решение, необходимо и достаточно, чтобы суммарные запасы поставщиков равнялись суммарным запросам потребителей, т.е. задача должна быть с правильным балансом.

Пример 1:

Составить математическую модель транспортной задачи перевоза груза из двух складов в 3 магазина:

Таблица 2

|  |  |  |  |
| --- | --- | --- | --- |
|   | 50 | 70 | 80 |
| 90 | 9 | 5 | 3 |
| 110 | 4 | 6 | 8 |

Решение. Введем переменные задачи(матрицу перевозок)

Запишем матрицу стоимостей

.

Целевая функция задачи равна сумме произведений всех соответствующих элементов матриц С и Х:

Данная функция, определяющая суммарные затраты на все перевозки, должна достигать минимального значения.

Составим систему ограничений задачи. Сумма всех перевозок, стоящих в первой строке матрицы Х, должна равняться запасам первого поставщика, а сумма перевозок во второй строке матрицы Х – запасам второго поставщика:

Это означает, что запасы поставщиков вывозятся полностью.

Суммы перевозок, стоящих в каждом столбце матрицы Ч, должны быть равны запросам соответствующих потребителей:

Это означает, что запросы потребителей удовлетворяются полностью.

Необходимо также учитывать, что перевозки не могут быть отрицательными:

 i=1,2,…,m; j=1,1,…,n.

Ответ: математическая модель задачи формулируется следующим образом: найти переменные задачи, обеспечивающие минимум функции

и удовлетворяющие системе ограничений

и условиям неотрицательности

 i=1,2,…,m j=1,2,…,n.

1.2 АЛГОРИТМ РЕШЕНИЯ ТРАНСПОРТНОЙ ЗАДАЧИ

1.2.1 СБАЛАНСИРОВАННОСТЬ ТРАНСПОРТНОЙ ЗАДАЧИ

Транспортная задача является сбалансированной, если суммарные запасы поставщиков равны суммарным запросам потребителей, т.е.

.

Если транспортная задача не сбалансирована, то возникают особенности в ее решении.

*Особенности решения транспортных задач с неправильным балансом:*

1.Если суммарные запасы поставщиков превосходят суммарные запросы потребителей, т.е.

то необходимо ввести фиктивного (n+1)-го потребителя с запросами равными разности суммарных запасов поставщиков и запросов потребителей, и нулевыми стоимостями перевозок единиц груза

2. Если суммарные запросы потребителей превосходят суммарные запасы поставщиков, т.е.

то необходимо ввести фиктивного (m+1)-го поставщика с запасами равные разности суммарных запросов потребителей и запасов поставщиков, и нулевыми стоимостями перевозок единиц груза

3. При составлении начального опорного решения в последнюю очередь следует распределять запасы фиктивного поставщика и удовлетворять запросы фиктивного потребителя, несмотря на то, что им соответствует наименьшая стоимость перевозок, равная нулю.

1.2.2 ОПОРНОЕ РЕШЕНИЕ ТРАНСПОРТНОЙ ЗАДАЧИ

Опорным решением транспортной задачи называется любое допустимое решение, для которого векторы условий, соответствующие положительным координатам, линейно независимы.

Ввиду того, что ранг системы векторов условий транспортной задачи равен N=m+n-1, опорное решение не может иметь отличных от нуля координат больше, чем N.

Для проверки линейной независимости векторов условий, соответствующих координатам допустимого решения, используют циклы.

Циклом называется такая последовательность клеток таблицы транспортной задачи в которой две и только соседние клетки расположены в одной строке или столбце, причем первая и последняя также находятся в одной строке или столбце.

Система векторов условий транспортной задачи линейно независима тогда и только тогда, когда из соответствующих им клеток таблицы нельзя образовать ни одного цикла. Следовательно, допустимое решение транспортной задачи , i=1,2,…,m; j=1,2,…,n является опорным только в том случае, когда из занятых им клеток таблицы нельзя образовать ни одного цикла.

Метод вычеркивания. Для проверки возможности образования цикла используется так называемый метод вычеркивания, который состоит в следующем.

Если в строке или столбце таблицы одна занятая клетка, то она не может входить в какой-либо цикл, так как цикл имеет две и только две клетки в каждом столбце. Следовательно, можно вычеркнуть все строки таблицы, содержащие по одной занятой клетке, затем вычеркнуть все столбцы, содержащие по одной занятой клетке, далее вернуться к строкам и продолжить вычеркивание строк и столбцов. Если в результате вычеркивания все строки и столбцы будут вычеркнуты, значит, из занятых клеток таблицы нельзя выделить часть, образующую цикл, и система соответствующих векторов условий является линейно независимой, а решение опорным. Если же после вычеркиваний останется часть клеток, то эти клетки образуют цикл, система соответствующих векторов условий линейно зависима, а решение не является опорным.

Метод минимальной стоимости. Данный метод позволяет построить опорное решение, которое достаточно близко к оптимальному, так как использует матрицу стоимостей транспортной задачи , i=1,2,…,m; j=1,2…,n. Данный метод состоит из ряда однотипных шагов, на каждом из которых заполняется только одна клетка таблицы, соответствующая минимальной стоимости , и исключается из рассмотрения только одна строка(поставщик) или один столбец(потребитель). Очередную клетку, соответствующую , заполняют также. Поставщик исключается из рассмотрения, если его запасы заканчиваются. Потребитель исключается из рассмотрения, если его запросы удовлетворены полностью. На каждом шаге исключается либо один поставщик, либо один потребитель. При этом если поставщик не исключен, но его запасы равны нулю, то на том шаге, когда от него требуется поставить груз, в соответствующую клетку таблицы заносится базисный нуль и лишь затем поставщик исключается из рассмотрения. Аналогично поступают с потребителем.

Пример 2:

Используя метод минимальной стоимости, построить начальное опорное решение транспортной задачи, доставки лекарств из трех складов в четыре аптеки.

Таблица 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|    | 80 | 120 | 160 | 120 |
| 120 | 1 | 3 | 4 | 2 |
| 160 | 4 | 5 | 8 | 3 |
| 200 | 2 | 3 | 6 | 7 |

Решение. Запишем отдельно матрицу стоимостей для того, чтобы удобнее было выбирать стоимости, вычеркивать строки и столбцы:

1 4 6 3

среди элементов матрицы стоимостей выбираем наименьшую стоимость . Это стоимость перевозки груза от первого поставщика первому потребителю. В соответствующую клетку (1,1) записываем максимально возможную перевозку (табл 4). Запасы первого поставщика уменьшаем на 80, . Исключаем из рассмотрения первого потребителя, так как его запросы удовлетворены. В матрице С вычеркиваем первый столбец.

Таблица 4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|    | 80 | 120 | 160 | 120 |
| 120 | 180 | 3 | 4 | 240 |
| 160 | 4 | 5 | 880 | 380 |
| 200 | 2 | 3120 | 680 | 7 |

В оставшейся матрицы С наименьшей является стоимость , максимально возможная перевозка, которую можно осуществить от первого поставщика к четвертому потребителю, равна . В соответствующую летку таблицы записываем перевозку . Запасы первого поставщика исчерпаны, исключаем его из рассмотрения. В матрице С вычеркиваем первую строку. Запросы четвертого потребителя уменьшаем на 40

В оставшейся части матрицы С минимальная стоимость . Заполняем одну из двух клеток таблицы (2,4) или (3,2). Пусть в клетку (2,4) запишем . Запросы четвертого потребителя удовлетворены полностью, исключаем его из рассмотрения, вычеркиваем четвертый столбец в матрице С. Уменьшаем запасы второго поставщика

В оставшейся части матрицы С минимальная стоимость . Запишем в клетку таблицы (3,2) перевозку Исключаем из рассмотрения второго потребителя, а из матрицы С второй столбец. Вычисляем

В оставшейся части матрицы С наименьшая стоимость Запишем в клетку таблицы (3,3) перевозку Исключаем из рассмотрения третьего поставщика, а из матрицы С третью строку. Определяем .

В матрице С остался единственный элемент . Записываем в клетку таблицы (2,3) перевозку .

Проверяем правильность построения опорного решения. Число занятых клеток таблицы равно N=m+n-1=3+4-1=6. Применяя метод вычеркивания, проверяем линейную независимость векторов условий, соответствующих положительным координатам решения. Порядок вычеркивания показан на матрице Х:

 1 2 5 6

Решение является «вычеркиваемым» и, следовательно, опорным.

Переход от опорного решения к другому. В транспортной задаче переход от оного опорного решения к другому осуществляется с помощью цикла. Для некоторой свободной клетки таблицы строится цикл, содержащий часть клеток, занятых опорным решением. По этому циклу перераспределяются объемы перевозок(осуществляется сдвиг по циклу). Перевозка «загружается» в выбранную свободную клетку и освобождается одна из занятых клеток, получается новое опорное решение.

Если таблица транспортной задачи содержит опорное решение, то для любой свободной клетки таблицы существует единственный цикл, содержащий эту клетку и часть клеток, занятых опорным решением.

Для удобства вычислений вершины циклов нумеруют и отмечают нечетные знаком «+», а четные знаком «-». Такой цикл называется означенным.

Сдвигом по циклу на величину называется увеличение объемов перевозок во всех нечетных клетках цикла, отмеченных знаком «+», и уменьшение объемов перевозок на ту же величину во всех не четных клетках, отмеченных знаком «-».

* + 1. МЕТОД ПОТЕНЦИАЛОВ

Широко распространенным методом решения транспортных задач является метод потенциалов.

Если допустимое решение , i=1,2,…,m; j=1,2,…n транспортной задачи является оптимальным, то существуют потенциалы (числа) поставщиков i=1,2,…,m и потребителей j=1,2,…,n, удовлетворяющее следующим образом:

Группа равенств (2.1) используется как система уравнений для нахождения потенциалов. Данная система уравнений имеет m+n неизвестных i=1,2,…,m и j=1,2,…,n. Число уравнений системы, как и число отличных от нуля координат невырожденного опорного решения, равно m+n-1. Так как число неизвестных системы на единицу больше числа уравнений, то одной из них можно задать значение произвольно, а остальные найти из системы.

Группа неравенств (2.2) используется для проверки оптимальности опорного решения. Эти неравенства удобнее представить в следующем виде:

 (2.3)

Числа называются оценками для свободных клеток таблицы (векторов условий) транспортной задачи.

Опорное решение является оптимальным, если для всех векторов условий (клеток таблицы) оценки неположительные.

Оценки для свободных клеток транспортной таблицы используются при улучшении опорного решения. Для этого находят клетку (l,k) таблицы, соответствующую . Если , то решение оптимальное. Если же , то для соответствующей клетки (l,k) строят цикл и улучшаю решение, перераспределяют груз

 по этому циклу.

*Алгоритм решения транспортных задач методом потенциалов:*

1. Проверить выполнение необходимого и достаточного условия разрешимости задачи. Если задача имеет неправильный баланс, то вводится фиктивный поставщик или потребитель с недостающими запасами или запросами и нулевыми стоимостями перевозок.
2. Построить начальное опорное решение (методом минимальной стоимости или каким-либо другим методом), проверить правильность его построения по количеству занятых клеток (их должно быть m+n-1) и убедиться в линейной независимости векторов условий (используется метод вычеркивания).
3. Построить систему потенциалов, соответствующих опорному решению. Для этого решают систему уравнений:

которая имеет бесконечное множество решений. Для нахождения частного решения системы одному из потенциалов (обычно тому, которому соответствует большее число занятых клеток) задают произвольно некоторое значение (чаще нуль). Остальные потенциалы однозначно определяются по формулам:

если известен потенциал , и

если известен потенциал

1. Проверить выполнения условия оптимальности для свободных клеток таблицы. Для этого вычисляют оценки для всех свободных клеток по формулам

и те из них, которые больше нуля, записываются в левые нижние углы клеток. Если для всех свободных клеток , то вычисляют значение целевой функции и решение задачи заканчивается, так как полученное решение является оптимальным. Если же имеется хотя бы одна клетка с положительной оценкой, опорное решение не является оптимальным.

5. Перейти к опорному решению, на котором значение целевой функции будет меньше. Для этого находят клетку таблицы задачи, которой соответствует наибольшая положительная оценка

Строят цикл, включающий в свой состав данную клетку и часть клеток, занятых опорным решением. В клетках цикла расставляют поочередно знаки «+» и «-», начиная с «+» в клетке с наибольшей положительной оценкой. Осуществляют сдвиг (перераспределение груза) по циклу на величину . Клетка со знаком «-», в которой достигается остается пустой. Если минимум достигается в нескольких клетках, то одна из них остается пустой, а в остальных проставляют базисные нули, чтобы число занятых клеток оставалось равным .

Далее перейти к пункту 3 данного алгоритма.

* + 1. МЕТОД СЕВЕРО-ЗАПАДНОГО УГЛА

Согласно данному методу запасы очередного поставщика используются для обеспечения запросов очередных потребителей до тех пор, пока не будут исчерпаны полностью, после чего используется запасы следующего по номеру поставщика.

Заполнение таблицы транспортной задачи начинается с левого верхнего угла и состоит из ряда однотипных шагов. На каждом шаге, исходя из запасов очередного поставщика и запросов очередного потребителя, заполняется только одна клетка и соответственно исключается из рассмотрения один поставщик или потребитель. При этом нулевые перевозки принято заносить в таблицу только в том случае, когда они попадают в клетку (i,j), подлежащую заполнению, т.е. в таблицу заносятся только базисные нули , остальные клетки с нулевыми перевозками остаются пустыми.

Во избежание ошибок после построения начального опорного решения необходимо проверить, что число занятых клеток равно m+n-1 и векторы условий, соответствующие этим клеткам, линейно независимы.

Необходимо иметь в виду, что метод северо-западного угла не учитывает стоимость перевозок, поэтому, опорное решение, построенное по данному методу, может быть далеким от оптимального.

Пример 3:

Составить опорное решение методом северо-западного угла транспортной задачи, в которой 5 поставщиков и 5 потребителей. данные записаны в таблице 6

Таблица 6

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|   | В150 | В240 | В330 | В420 | В510 |
| А110 |  |  |  |  |  |
| А220 |  |  |  |  |  |
| А330 |  |  |  |  |  |
| А440 |  |  |  |  |  |
| А550 |  |  |  |  |  |

Решение:

Распределяем запасы первого поставщика. Так как его запасы меньше запросов первого потребителя , то в клетку (1,1) записываем перевозку и исключаем из рассмотрения первого поставщика. Определяем оставшиеся неудовлетворенными запросы первого потребителя .

Распределяем запасы второго поставщика. Так как его запасы , меньше запросов первого потребителя , то записываем в клетку (2,1) перевозку и исключаем из рассмотрения второго поставщика. Определяем оставшиеся неудовлетворенными запросы первого потребителя .

Распределяем запасы третьего поставщика . Так как его запасы больше запросов первого потребителя , то записываем в клетку (3,1) перевозку и исключаем из рассмотрения первого потребителя. Определяем оставшиеся неудовлетворенными запросы третьего поставщика .

Распределяем запасы третьего поставщика . Так как его запасы меньше запросов второго потребителя , то в клетку (3,2) записываем перевозку и исключаем из рассмотрения третьего поставщика. Определяем оставшиеся неудовлетворенными запросы второго потребителя .

Распределяем запасы четвертого поставщика . Так как его запасы больше запросов второго потребителя , то записываем в клетку (4,2) перевозку и исключаем из рассмотрения второго потребителя. Определяем оставшиеся неудовлетворенными запросы четвертого поставщика .

Распределяем запасы четвертого поставщика . Так как его запасы меньше запросов третьего потребителя , то в клетку (4,3) записываем перевозку и исключаем из рассмотрения четвертого поставщика. Определяем оставшиеся неудовлетворенными запросы третьего потребителя .

Распределяем запасы пятого поставщика. Так как его запасы больше запросов третьего потребителя , то в клетку (5,3) записываем перевозку и исключаем из рассмотрения третьего потребителя. Определяем оставшиеся неудовлетворенными запасы пятого поставщика.

Распределяем запасы пятого поставщика. Так как его запасы больше запросов четвертого потребителя , то в клетку (5,4) записываем перевозку и исключаем из рассмотрения четвертого потребителя. Определяем оставшиеся неудовлетворенными запасы пятого поставщика.

Распределяем запасы пятого поставщика. Так как его запасы равны запросам пятого потребителя , то в клетку (5,5) записываем перевозку и исключаем из рассмотрения пятого поставщика и пятого потребителя.

Ввиду того, что задача с правильным балансом, запасы всех поставщиков исчерпаны и запросы всех потребителей удовлетворены.

Результаты построения опорного решения приведены в таблице 7.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|   | В150 | В240 | В330 | В420 | В510 |
| А110 | 10 | - | - | -  | - |
| А220 | 20 | - | - | - | - |
| А330 | 20 | 10 | - | - | - |
| А440 | - | 30 | 10 | - | - |
| А550 | - | - | 20 | 20 | 10 |

1.3 БЛОК-СХЕМА (АЛГОРИТМ РЕШЕНИЯ)

2

Ввод

данных

3

Пров. ввода

6

Запасы меньше потроеб

4

Суммирование потребностей потребителей

5

Суммирование запасов поставщиков

 7

Добавл. Мнимого поставщика

8

 Добавл. Мнимого потребителя

 нет нет

 да

13

цикл

14

построение нового решения

9

Построение начального плана

10

Нахождение оценок свободных клеток

11

Оценки больше 0

12

Оптимальное решение

 Метод

 северо-

 - западного

 угла

 метод

 потенциалов

2. ФОРМЫ ВХОДНОЙ ИНФОРМАЦИИ

Входные данные вводятся с клавиатуры

* Запасы i-го поставщика
* запросы j-го потребителя

В данном примере

* запасы поставщиков(10; 20; 30; 40; 50)
* запросы потребителей(50; 40; 30; 20; 10)

3. ФОРМЫ ВЫХОДНОЙ ИНФОРМАЦИИ

Информация выводится на экран в виде таблицы с введенными данными и допустимый начальный базис

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|   | В150 | В240 | В330 | В420 | В510 |
| А110 | 10 | - | - | -  | - |
| А220 | 20 | - | - | - | - |
| А330 | 20 | 10 | - | - | - |
| А440 | - | 30 | 10 | - | - |
| А550 | - | - | 20 | 20 | 10 |

4. ИНСТРУКЦИЯ ДЛЯ ПОЛЬЗОВАТЕЛЯ

**Общие сведения:**

Программа производит вычисления допустимого начального базиса

Задача является сбалансированной. Поиск начального базиса происходит методом «северо-западного угла»

**Управление:**

Данные вводятся с клавиатуры:

Пользователь вводит запасы i-го потребителя. После нажатия клавиши «0» пользователь вводит запросы j-го поставщика. Далее на экране после нажатия клавиши «Enter» появляется таблица с вводимыми данными и начальный базис.

5. ИНСТРУКЦИЯ ДЛЯ ПРОГРАММИСТА

Данная программа реализуется с помощью процедурного языка Turbo Pascal 7.0, используется текстовый режим.

5.1 ТРЕБУЕМЫЕ ИНФОРМАЦИОННО–ВЫЧИСЛИТЕЛЬНЫЕ СРЕДСТВА:

1. техническое обеспечение: IBM PC\XT совместимые машины

а) оперативная память – не менее 8Мб

б) свободное место на жестком диске – не менее 60Кб

в) центральный процессор – от Intel 8088 до семейства Pentium или совместимых с ним

2) информационные средства для нормального функционирования программы достаточно иметь информационную систему MS DOS

5.2 ТИПЫ ПЕРЕМЕННЫХ, ИСПОЛЬЗОВАННЫХ В ПРОГРАММЕ:

const n=20 (строки)

m=20 (столбцы)

a:array [1..n] of integer; {массив запасов}

b:array [1..m] of integer; {массив потребностей}

a1:array [1..n] of integer; {вспомогательный массив запасов}

b1:array [1..m] of integer; {вспомогательный массив потребностей}

c:array [1..n,1..m] of integer; {основной массив в который производится запись базисного решения}

i,j,k,x,y,s1,s2:integer;

5.3 ПРОЦЕДУРЫ

procedure vvod\_klav;(ввод данных с клавиатуры)

begin

i:=1;

k:=0;

s1:=0;

while (k=0) and (i<n) do

 begin

 write('введите запaсы ',i,'-того поставщика: ');

 readln(a[i]);

 if a[i]=0 then

 begin

 k:=1;

 i:=i-1;

 end

 else

 begin

 a1[i]:=a[i];

 s1:=s1+a1[i];

 i:=i+1;

 end;

 end;

j:=1;

k:=0;

s2:=0;

textcolor(5);

while (k=0) and (j<m) do

 begin

 write('введите запрос ',j,'-того потребителя: ');

 readln(b[j]);

 if b[j]=0 then

 begin

 k:=1;

 j:=j-1;

 end

 else

 begin

 b1[j]:=b[j];

 s2:=s2+b1[j];

 j:=j+1;

 end;

 end;

textcolor(yellow);

k:=0;

if s1<s2 then

 begin

writeln('ошибка ввода, проверьте баланс');

 readln;

 halt;

 end;

if (s2<s1) and (k=0) then

 begin

writeln('ошибка ввода, проверьте баланс');

 readln;

 halt; end;

x:=i;

y:=j;

end;

ЗАКЛЮЧЕНИЕ

Мне была поставлена задача составить программу для расчета начального базиса сбалансированной транспортной задачи, где суммарные запасы поставщиков равны суммарным запросам потребителей.

Программа реализована на языке программирования Паскаль.

Все вводимые данные и начальный базис выводятся в виде таблицы.

В программе удобный и понятный пользовательский интерфейс. Для ввода данных используется клавиатура. Данные, выводимые программой, соответствуют тем, что получены при расчетах без использования компьютера. Таким образом, поставленная задача была выполнена.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 Общий курс высшей математики для экономистов. Учебник / под ред В.И. Ермакова.- М.: ИНФА – М. – 656 с. – (серия «высшее образование»).

2 Сборник задач и упражнений по высшей математике: математическое программирование: учебник пособие / А.В. Кузнецов, В.А. Сакович, Н.И. Холод и др; МН.: выш. ик., 2002. – 447с.:ил.

3 Т.Л. Партыкина, И.И. Попов Математические методы: учебник. – М.: ФОРУМ: ИНФА-М, 2005. – 464 с.: ил – (профессиональное образование)

4. И.Г. Семакин Основы программирования: учебник для сред. проф. Образования / И.Г. Семакин, А.П.Шестаков. – 2-е изд., стер,- М.: Издательский центр «Академия», 2003.-432 с.

5 Федосеев В.В. и др. Экономико-математические методы и прикладные модели: учебное пособие для ВУЗов. - М.: Юнити, 2002.

6 Коршунов Ю.М. математические основы кибернетики: учебное пособие для ВУЗов. – М.: Энергоатомиздат, 1987.

**ПРИЛОЖЕНИЕ А**

Листинг программы

program sev\_zap;

uses crt; {подключение модуля "crt"}

const n=5; {количество строк}

 m=5; {количество столбцов}

var a:array [1..n] of integer; {массив запасов}

 b:array [1..m] of integer; {массив потребностей}

 a1:array [1..n] of integer; {вспомогательный массив запасов}

 b1:array [1..m] of integer; {вспомогательный массив потребностей}

 c:array [1..n,1..m] of integer; {основной массив в который производится запись базисного решения}

 i,j,k,x,y,s1,s2:integer;

{ввод с клавиатуры}

procedure vvod\_klav;

begin

i:=1;

k:=0;

s1:=0;

while (k=0) and (i<n) do

 begin

 write('введите запaсы ',i,'-того поставщика: ');

 readln(a[i]);

 if a[i]=0 then

 begin

 k:=1;

 i:=i-1;

 end

 else

 begin

 a1[i]:=a[i];

 s1:=s1+a1[i];

 i:=i+1;

 end;

 end;

j:=1;

k:=0;

s2:=0;

textcolor(5);

while (k=0) and (j<m) do

 begin

 write('введите запрос ',j,'-того потребителя: ');

 readln(b[j]);

 if b[j]=0 then

 begin

 k:=1;

 j:=j-1;

 end

 else

 begin

 b1[j]:=b[j];

 s2:=s2+b1[j];

 j:=j+1;

 end;

 end;

textcolor(yellow);

k:=0;

if s1<s2 then

 begin

writeln('ошибка ввода, проверьте баланс');

 readln;

 halt;

 end;

if (s2<s1) and (k=0) then

 begin

writeln('ошибка ввода, проверьте баланс');

 readln;

 halt;

 end;

x:=i;

y:=j;

end;

begin

textcolor(white);

clrscr; {очистка экрана}

writeln(‘Построение начального базиса в сбалансированной транспортной задаче методом северо-западного угла’);

writeln;

writeln(‘Программу составил: Руднев Егор Николаевич’);

writeln;

vvod\_klav; {процедура ввода с клавиатуры}

repeat

k:=0;

if (b[j]-a[i]<0) then

 begin

 c[i,j]:=b[j];

 a[i]:=a[i]-b[j];

 b[j]:=0;

 j:=j-1;

 k:=1;

 end;

if (b[j]-a[i]>0) and (k=0) then

 begin

 c[i,j]:=a[i];

 b[j]:=b[j]-a[i];

 a[i]:=0;

 i:=i-1;

 k:=1;

 end;

if (b[j]-a[i]=0) and (k=0) then

 begin

 c[i,j]:=a[i];

 a[i]:=0;

 b[j]:=0;

 i:=i-1;

 j:=j-1;

 end;

if (i=0) or (j=0) then break;

until false;

{вывод на экран базисного решения}

clrscr;

textcolor(white);

for i:=1 to x do

 begin

 for j:=1 to y do

 if j=y then write(c[i,j]:6,' │ ',a1[i])

 else

 write(c[i,j]:6);

 writeln;

 end;

write(' ');

for i:=1 to y\*6-4 do

 write(#196);

writeln('┘');

for j:=1 to y do

 write(b1[j]:6);

readln;

end.