МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования  
**«Национальный исследовательский   
Нижегородский государственный университет им. Н. И. Лобачевского» (ННГУ)**

**Институт информационных технологий, математики и механики**

Направление подготовки:

«Фундаментальная информатика и информационные технологии»

**ОТЧЕТ**

по лабораторной работе

**ОПТИМАЛЬНЫЙ ПЛАН ПЕРЕРАБОТКИ САХАРНОЙ СВЁКЛЫ**

**Выполнили:** студенты группы

382006-2

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д.А. Варызгин

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д.А. Збруев

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.В. Кротов

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н.Р Кудрявцев

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н.В. Смирнов

**Проверил:** к.ф.-м.н., доц.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А. И. Эгамов

Нижний Новгород  
2023

Содержание

[1. Введение 3](#_Toc132641789)

[2. Постановка задачи 4](#_Toc132641790)

[3. Теоретическая часть 4](#_Toc132641791)

[**3.1 Задача о назначениях. Венгерский алгоритм. Метод Мака.** 4](#_Toc132641792)

[**3.2 Жадный и бережливый алгоритмы.** 7](#_Toc132641793)

[4. Компьютерная часть 20](#_Toc132641794)

[5. Руководство пользователя. Оценка результатов экспериментов. Дополнительное задание. 23](#_Toc132641795)

[6. Заключение 27](#_Toc132641796)

[7. Список литературы 28](#_Toc132641797)

1. Введение

Сахарная промышленность нуждается в новых программных средствах, позволяющих дать рекомендации для построения оптимальной стратегии переработки. Эта проблема часто затрагивается в работах, посвященных пищевой промышленности и промышленности, перерабатывающей сельскохозяйственные продукты. При хранении различные партии сырья с течением времени, вообще говоря, по-разному снижают свою производственную ценность (снижается доля полезного ингредиента, например, для сахарной свеклы – это доля сахара в сахарной свекле. Поэтому оптимальный график переработки партий сырья может существенно повысить итоговый выход конечного продукта. Выход готового продукта (сахара) зависит от многих величин, от процентного содержания грязи на свекле, содержания нитратов, повреждений при транспортировке на переработку, температуры обработки и 5 так далее, но, тем не менее, основное влияние на выход конечного продукта оказывает процентное содержание сахарозы. Для решения этой важной прикладной задачи необходимо глубокое знание алгоритмов дискретной оптимизации, владение современными программными средствами и способность оценивать полученные путем численных расчетов результаты с точки зрения их прикладной значимости. Таким образом, данная проблема сочетает все необходимые компоненты для выполнения проектных работ, представляет хороший материал для обучения студентов в области ИТ и может служить основой для создания различных проектов в рамках учебной дисциплины.

1. Постановка задачи

Необходимо построить оптимальную стратегию переработки свеклы.

3. Теоретическая часть

## **3.1 Задача о назначениях. Венгерский алгоритм. Метод Мака.**

**Задача о назначениях.**

В процессе управления производством часто возникают задачи, связанные с назначением исполнителей на различные виды работ. Например, подбор кадров и назначение кандидатов на вакантные должности, распределение источников капитальных вложений между различными проектами научно-технического развития, распределение экипажей самолетов между авиалиниями и т.д. Задача о назначениях (ЗН) является специфической задачей транспортного типа дискретного линейного программирования. При рассмотрении ЗН можно использовать теорию, методологию и результаты линейного программирования, а для решения применять и симплекс-метод, и метод потенциалов, и методы отсечения, и методы ветвей и границ. Однако эти методы в данном случае неэффективны, так как любое допустимое базисное решение задачи о назначениях (в силу специфики задачи) является вырожденным. Оригинальным алгоритмом решения ЗН является венгерский метод, который был разработан и опубликован в 1955 году Харолдом Куном. Ему же принадлежит и название, данное в честь венгерских математиков Кенига и Эгервари. В данной работе рассматривается исходный вариант метода как поучительный пример интересных и необычных математических методов. Учебное пособие предназначено для студентов экономических специальностей. Оно может быть интересно всем интересующимся теорией принятия оптимальных управленческих решений.

**Венгерский алгоритм.**

Венгерский алгоритм — алгоритм [оптимизации](https://dic.academic.ru/dic.nsf/ruwiki/213254), решающий задачу о назначениях за полиномиальное время (см. [исследование операций](https://dic.academic.ru/dic.nsf/ruwiki/30517)). Он был разработан и опубликован Харолдом Куном в [1955 году](https://dic.academic.ru/dic.nsf/ruwiki/1405). Автор дал ему имя «венгерский метод» в связи с тем, что алгоритм в значительной степени основан на более ранних работах двух [венгерских](https://dic.academic.ru/dic.nsf/ruwiki/986) математиков ([Кёнига](https://dic.academic.ru/dic.nsf/ruwiki/1539344) и Эгервари).

Венгерский алгоритм применим в таких случаях, как:

Задача о назначении работников на должности. Необходимо распределить работников на должности так, чтобы достигалась максимальная эффективность, или были минимальные затраты на работу.

Назначение машин на производственные секции. Распределение машин так, чтобы при их работе производство было максимально прибыльным, или затраты на их содержание минимальны.

Выбор кандидатов на разные вакансии по оценкам. Этот пример разберем ниже.

**Метод Мака**

Метод Мака применяется для распределения функций между элементами системы, когда каждый из этих элементов способен выполнять любую из функций с разной эффективностью. Предполагается выполнение одним элементом одной функции, и на выходе системы необходимо получить максимальную эффективность. К примеру, в организации n работников, между которыми требуется распределить n заданий так, чтобы все задания были выполнены за суммарно минимальное время. Время выполнения заданий напрямую зависит от свойств характера исполнителей, их квалификации. Следовательно, эффективность труда исполнителей, в силу этих и иных причин, будет различна. Данная задача носит название задачи о назначениях и представляет собой частный случай более общей транспортной задачи. Существует определенный алгоритм решения задачи о назначениях методом Мака. В первую очередь, составляется матрица: допустим, по вертикали указываются номера работников (Р) организации, а по горизонтали – номера заданий (З), которые соответствуют разным должностям. Таблицу заполняем значениями времени выполнения заданий. Требуется найти минимальное суммарное время выполнения всех n заданий.

## **3.2 Жадный и бережливый алгоритмы.**

Жадными называют класс алгоритмов, заключающихся в принятии локально оптимальных решений на каждом этапе. Локально оптимальное решение просто вычислить.

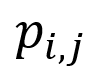
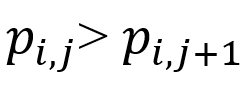
В некоторых случаях жадные алгоритмы приводят к оптимальным конечным решениям, а в других - нет. Доказать корректность жадных алгоритмов часто бывает сложно.

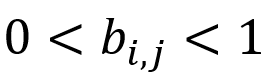
**Описание алгоритма для нашей задачи**

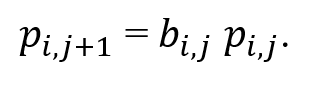
Имеем n партий сахарной свёклы равной массы, они пронумерованы от 1 до n, каждая имеет начальную сахаристость (доля сахара). На обработку одной партии уходят одни сутки, за одни сутки можно обработать только одну партию. Если партию не обработать, она теряет долю сахара.

Представим в виде матрицы:

P (n×n) – матрица сахаристости. Строки: партии, столбцы: сутки.

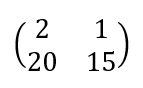
– сахаристость i-ой партии в j-е сутки. Так как она убывает, .

Коэффициент деградации: . Определяют потерю сахаристости i-ой партии в j-е сутки:



**Жадный** алгоритм здесь заключается в том, что на переработку идёт партия с **наибольшей** сахаристостью на данный момент.

Простой пример (в условных единицах):

 Строки: партии, столбцы: сутки

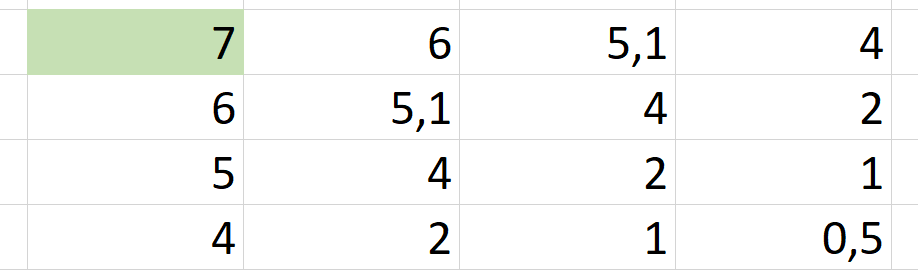
Жадный алгоритм: берём 20 в первые сутки, во вторые сутки остаётся 1.

**Итого: 21.**

**Бережливый** алгоритм отличается от жадного, тем, что на переработку идёт партия с **наименьшей** сахаристостью на данный момент.

**3.3 Порядок переработки 4 партий с помощью уже известных алгоритмов**

***Задача 1.***  **Дана матрица P1:**



Провести следующие стратегии переработки:

**а)** Венгерским алгоритмом (максимум целевой функции)

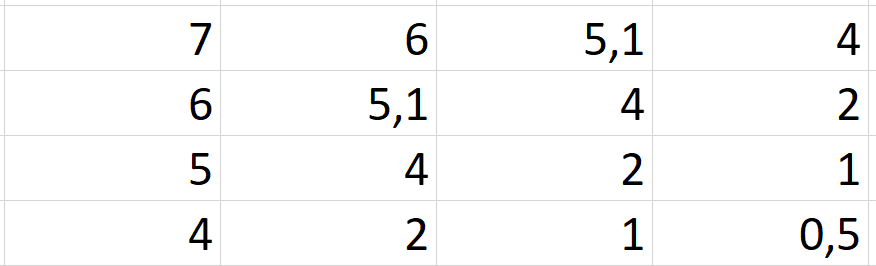
**б)** Венгерским алгоритмом (максимум целевой функции)

**в)** Жадным алгоритмом

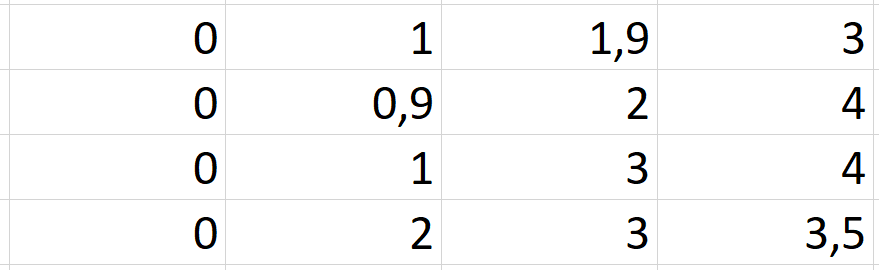
**г)** Бережливым алгоритмом

**Решение:**

**а) Венгерским алгоритмом (максимум целевой функции)**

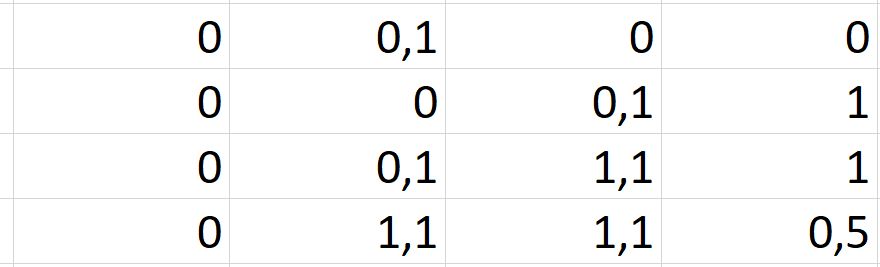


1) Вычитаем из каждой строки её максимальный элемент и умножаем полученную матрицу на -1



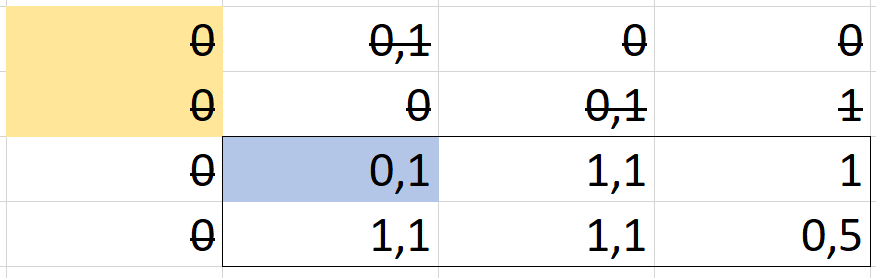
2) Вычитаем минимальный элемент каждой строки из этой строки, это 0, ничего не изменится.

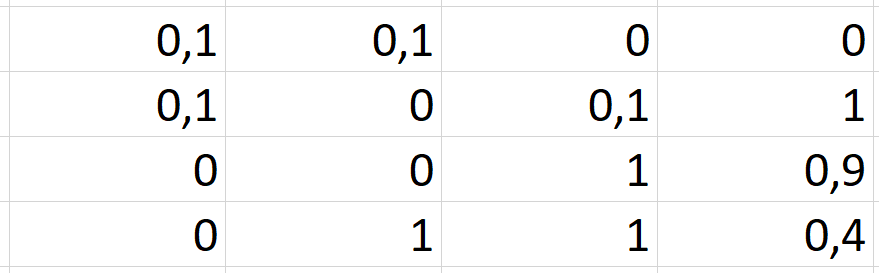
Вычитаем минимальный элемент каждого столбца из этого столбца.



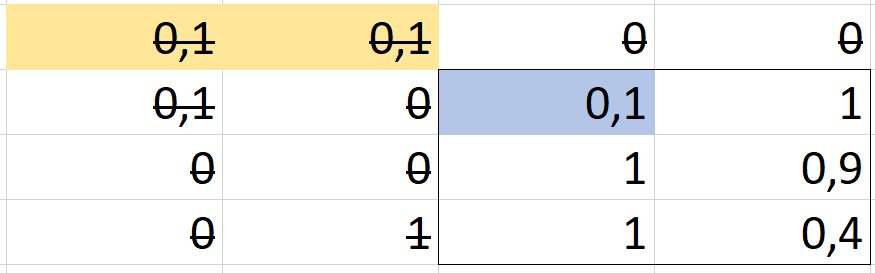
3) Вычеркнем стоки и столбцы, которые содержат нули. Количество удалённых столбцов и строк должно быть минимальным.

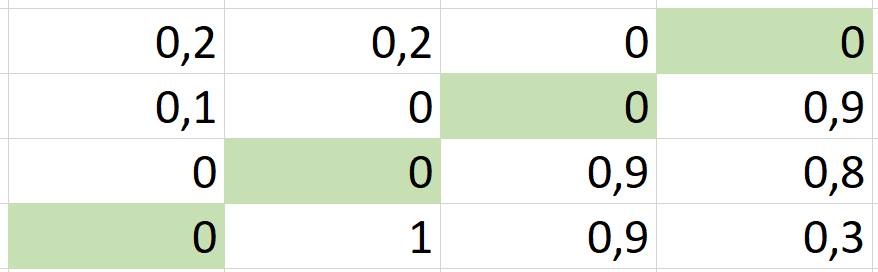
Вычитаем минимальный элемент получившейся матрицы из незачёркнутых и прибавляем его к элементам, которые расположены на пересечении удалённых строк и столбцов.



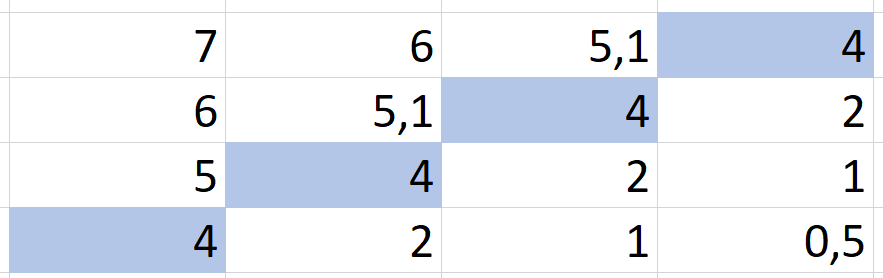


4) Так как в полученной матрице мы не можем выбрать нули так, чтобы они все были из разных строк и столбцов, повторим шаги 2 и 3.





5) Получилось выбрать нули так, чтобы они все были из разных строк и столбцов. Соответствующие им индексы в исходной матрице являются решением задачи.



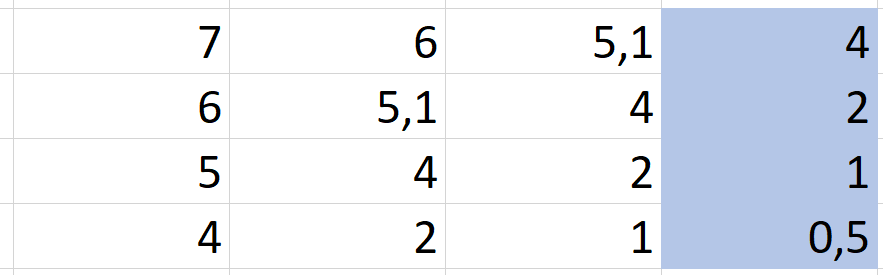
Их сумма максимальна среди всех наборов элементов этой матрицы: 16.

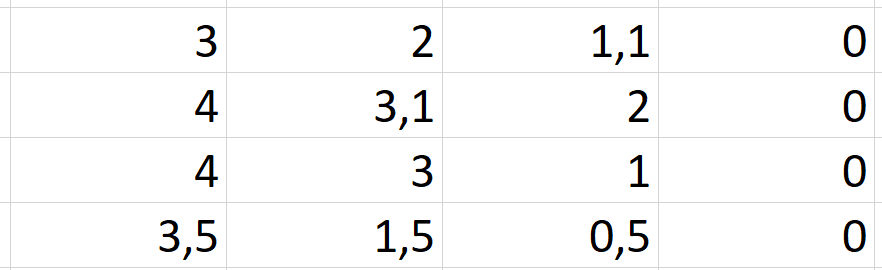
**Результат: 16**

**б) Венгерским алгоритмом (максимум целевой функции):**

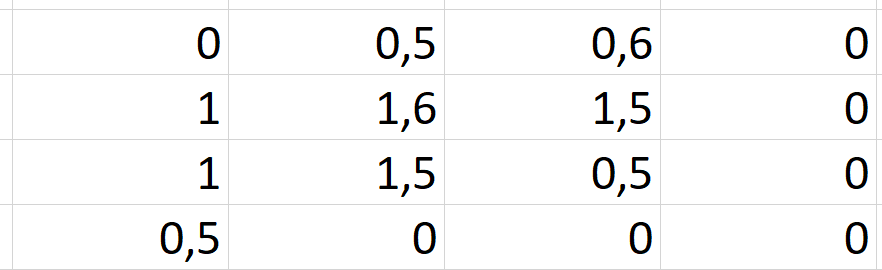
1) Венгерский алгоритм на минимум отличается только тем, что пропускаем первый шаг решения задачи на максимум.

2) Вычитаем минимальный элемент каждой строки из этой строки:

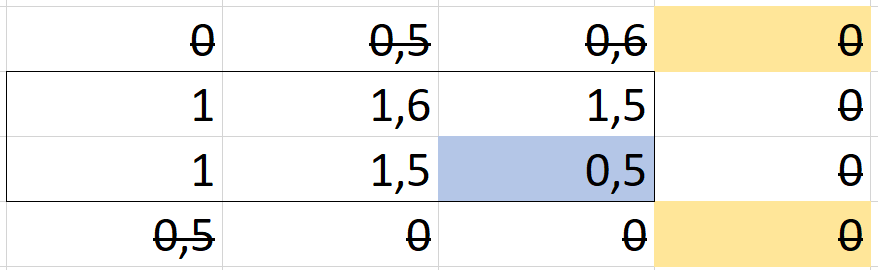


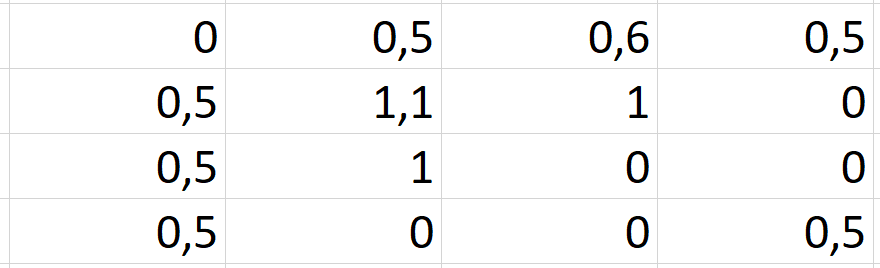


Теперь то же со столбцами:

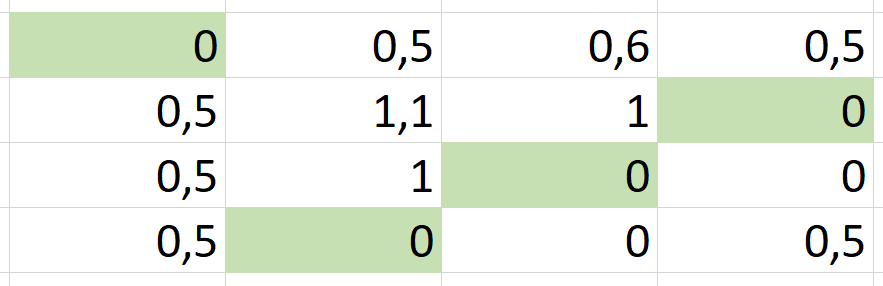


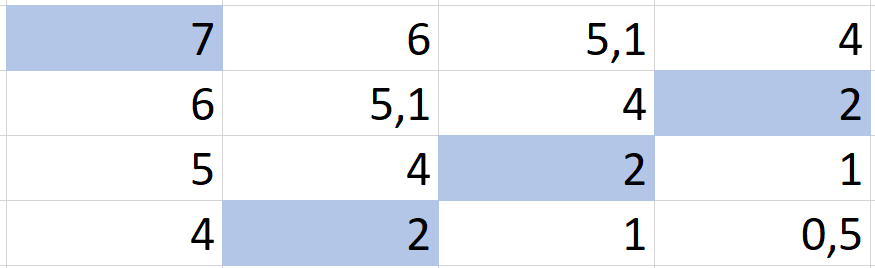
3)





4)

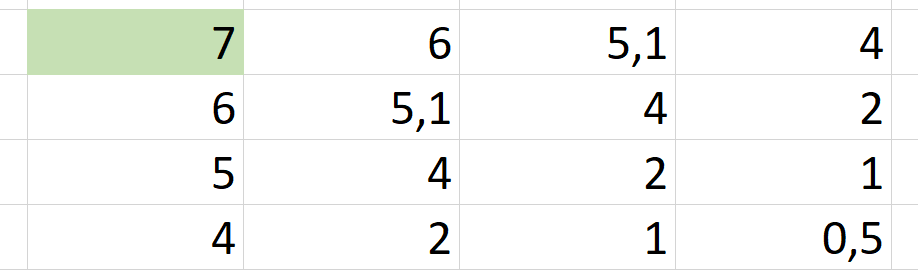




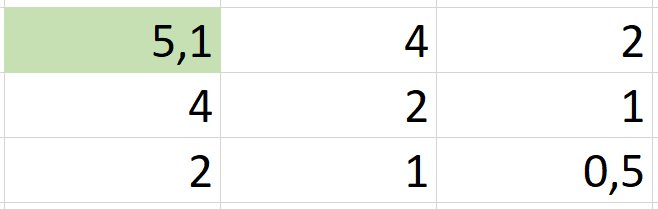
**Результат: 13.**

**в) Жадным (наибольшая сахаристость на данном этапе)**

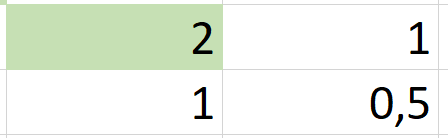
1)



2)



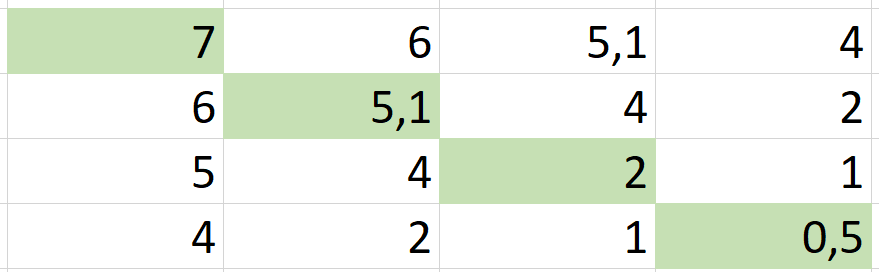
3)



4)



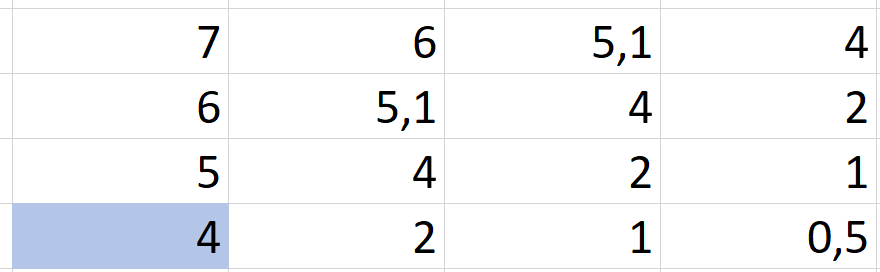
Итог:



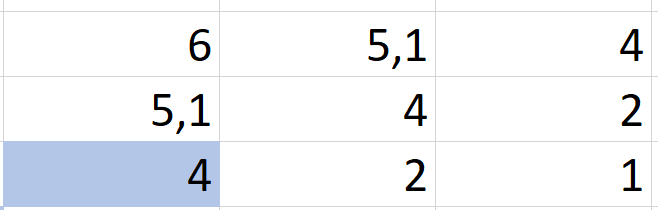
**Результат: 14,6.**

**г) Бережливым (наименьшая сахаристость на данном этапе)**

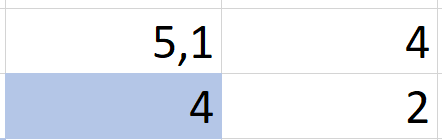
1)



2)



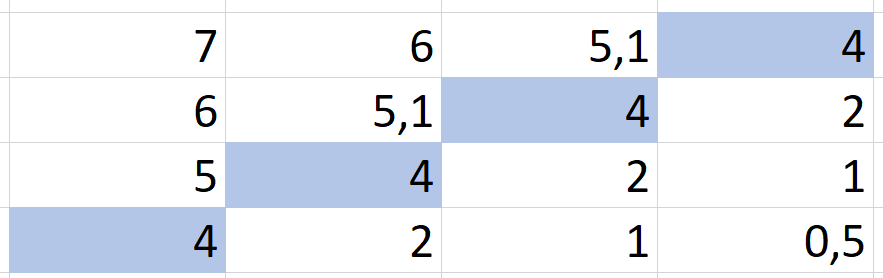
3)



4)



Итог:



**Результат: 16**

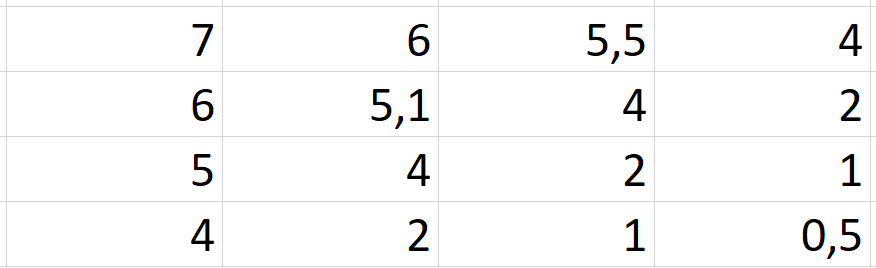
***Задача 2.* Осуществить те же операции для матрицы P2, которая получается из матрицы P1 заменой p1,3 на 5,5.**

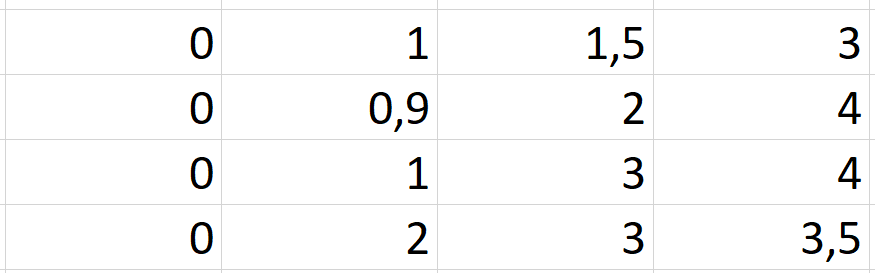
**Решение:**

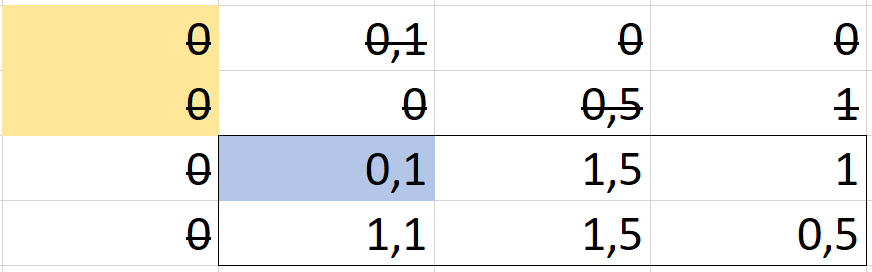
Замена p1,3 = 5,1 на 5,5 **не повлияет на решение жадного и бережливого** **алгоритмов**, так как p1,3 будет «вычеркнута» на первом этапе в ходе Жадного алгоритма и на третьем этапе в ходе Бережливого.

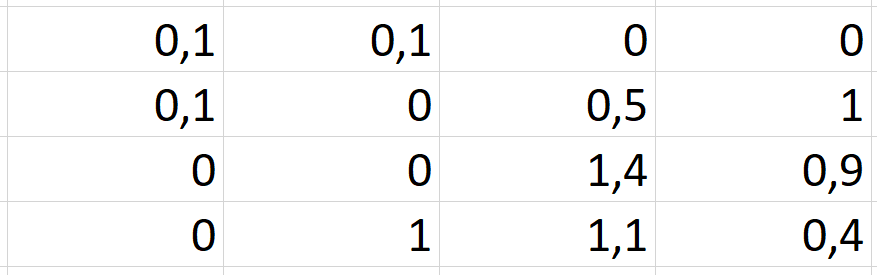
**Результаты** для них те же: **14, 6 и 16.**

Венгерский алгоритм на максимум:

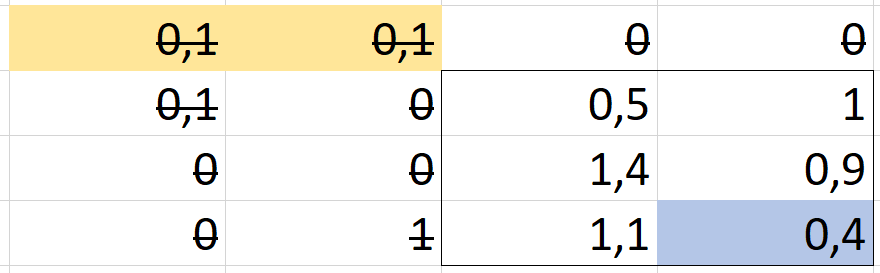


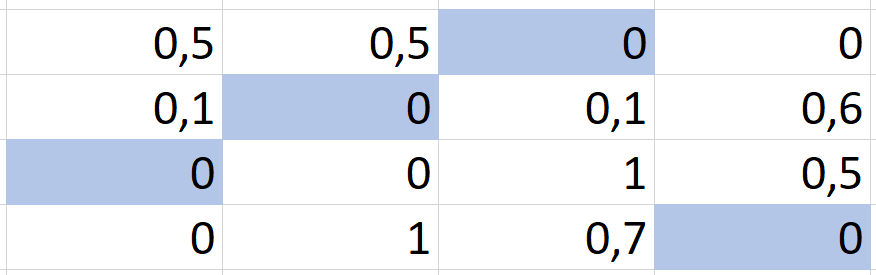


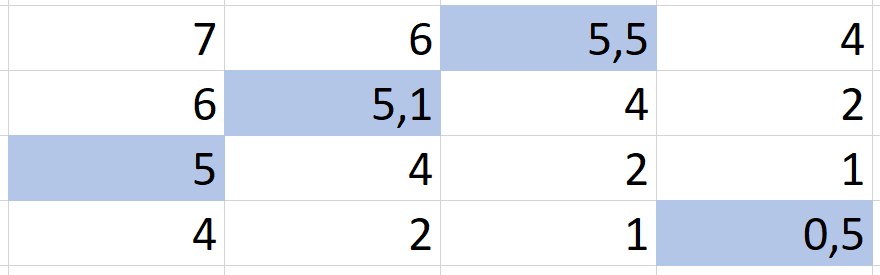




Нельзя выбрать нули, повторим шаги 2 и 3:







**Результат: 16,1.**

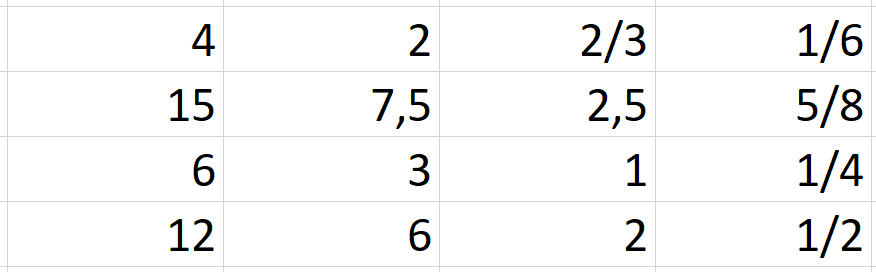
Венгерский алгоритм на минимум:

Решение задачи на минимум не поменялось по сравнению с матрицей P1.

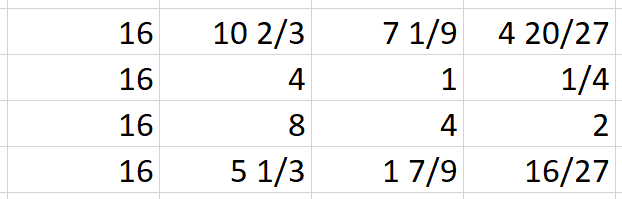
**Результат: 13.**

***Задача 3.* Даны матрицы P3 и P4:**

***P3:***

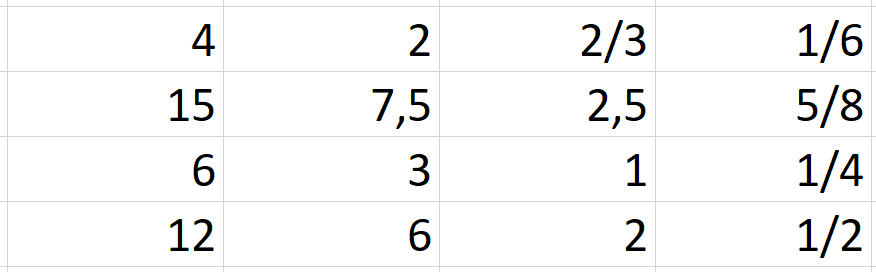


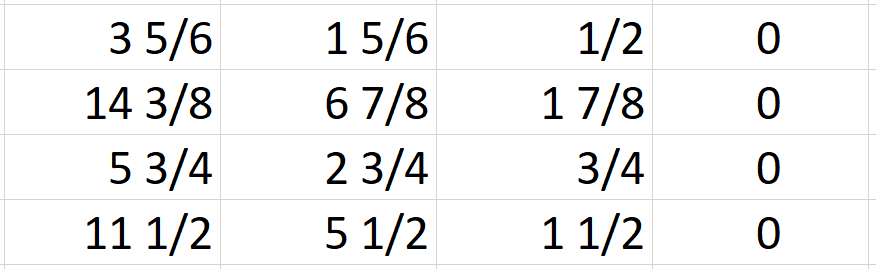
**P4:**

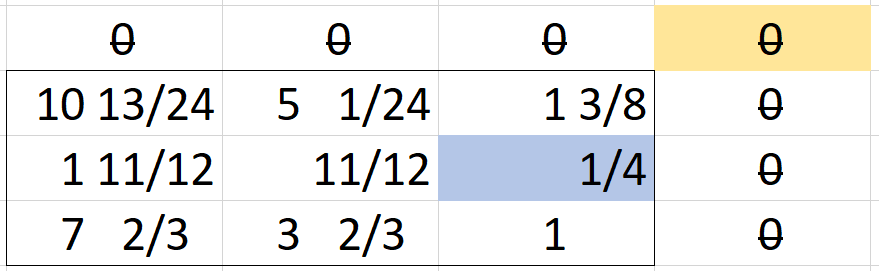


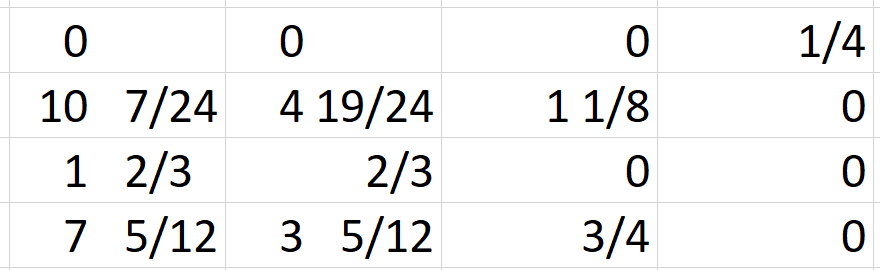
Решить задачу о назначениях.

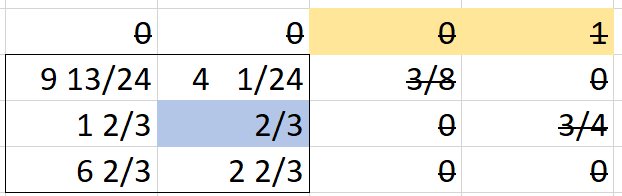
**Решение для P3:**

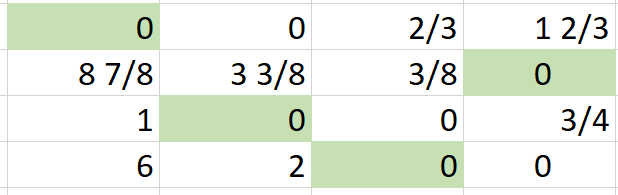






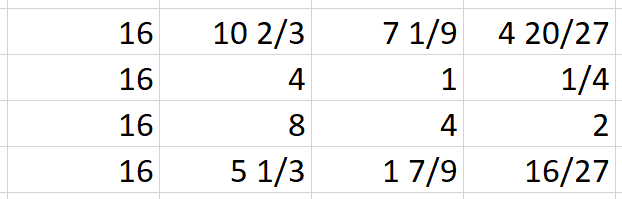


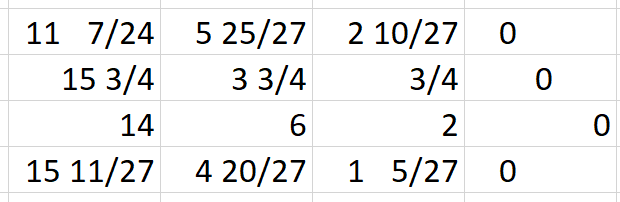


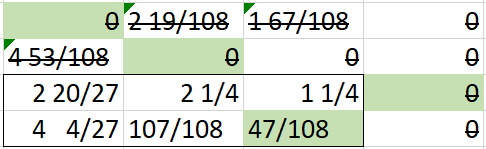


**min** = 4 + 3 + 2 + 5/8 = **9,625**

**Решение для P4:**



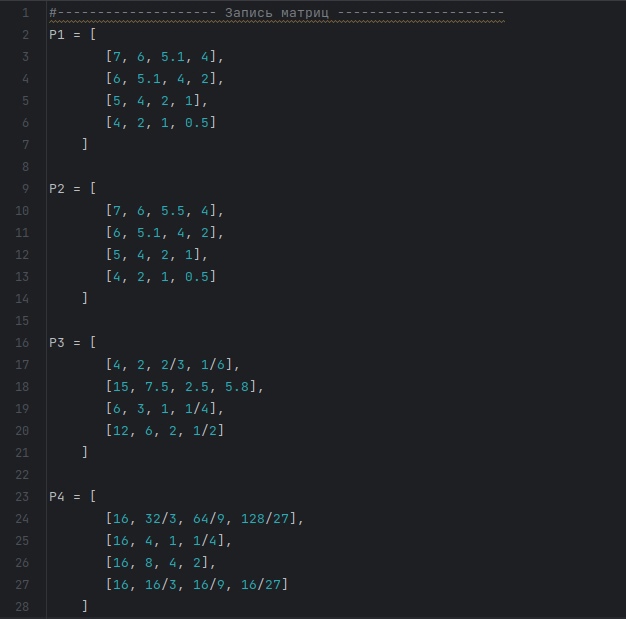


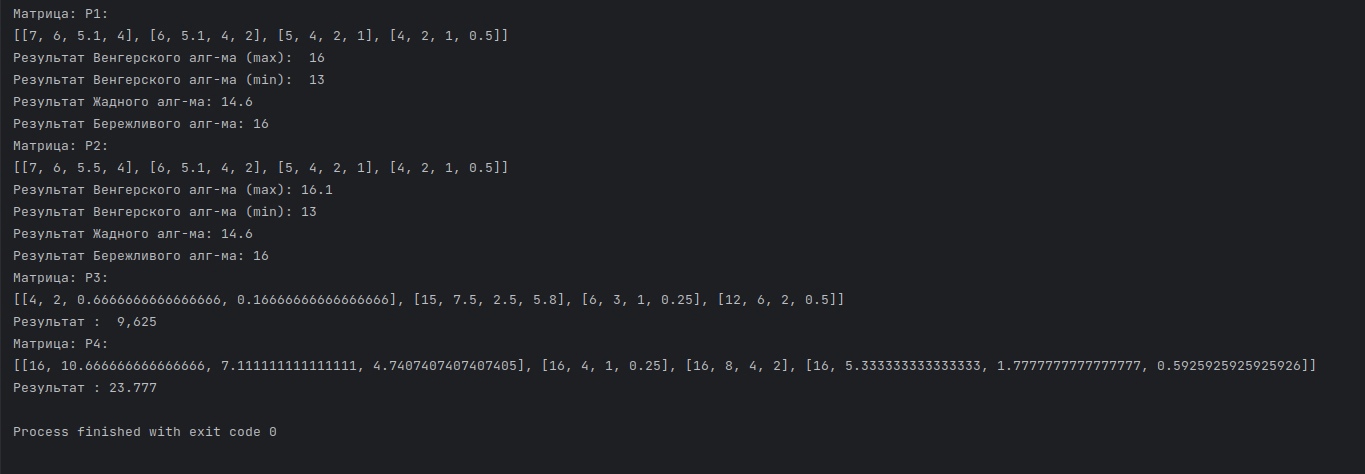


**min** = 16 + 4 + 16/9 + 2 = **23 7/9**

4. Компьютерная часть

Для задач из теоретической части найдем результаты работы алгоритмов. Для этого создадим матрицы P1, P2, P3, P4 с такими же значениями.



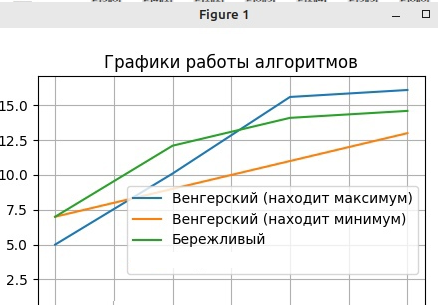
После вызова функций наших алгоритмов получим следующие результаты.

Видим, что результаты теоретической и практической частей совпадают.

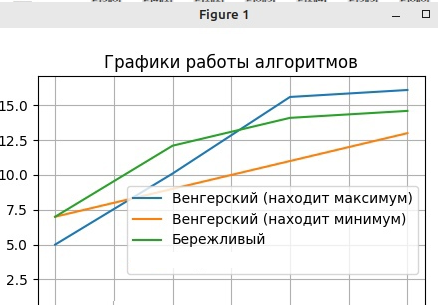
Теперь построим графики, отражающие решения задач для матриц

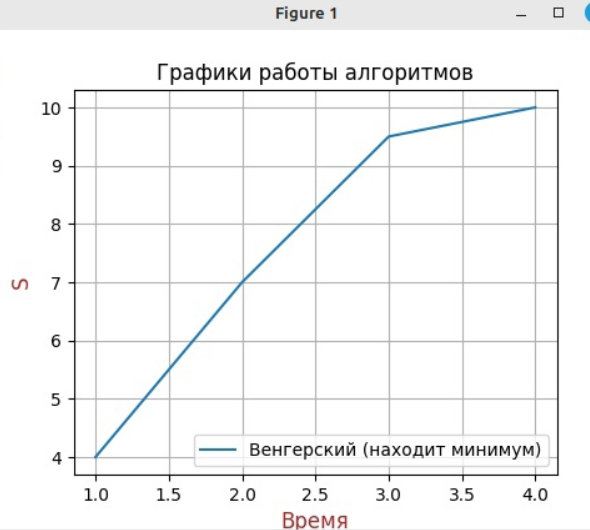
*P1, P2, P3, P4.*

Для P1:

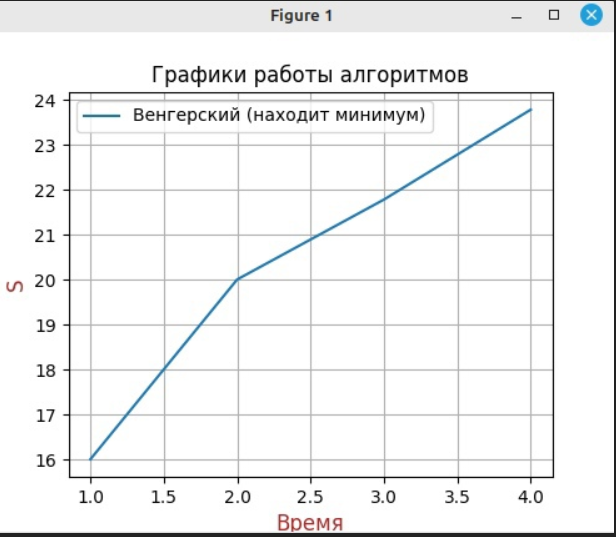


Для P2:



Для P3:

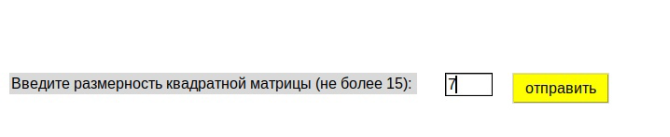
Для P4:

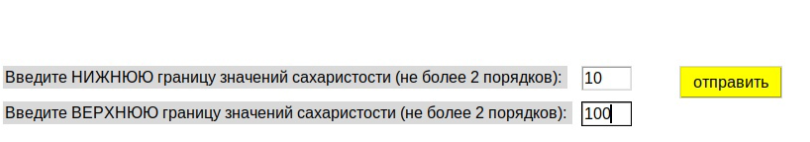


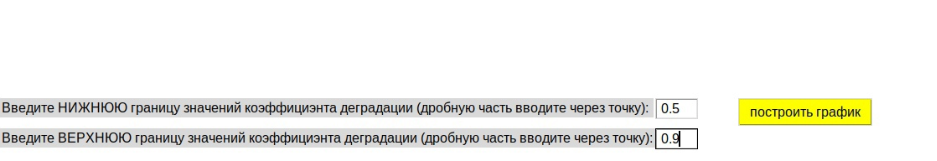
Нетрудно заметить, что полученные с помощью компьютерной программы графики соответствуют значениям, полученным с помощью алгоритмов.

5. Руководство пользователя. Оценка результатов экспериментов. Дополнительное задание.

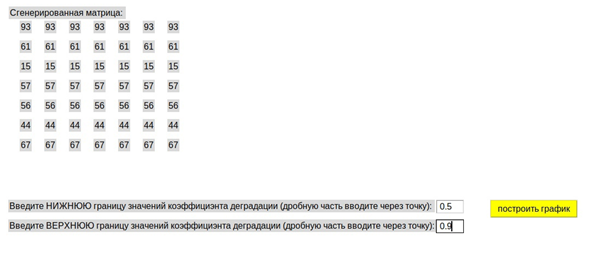
При запуске приложения пользователю предлагается ввести размерность матрицы сахаристости.

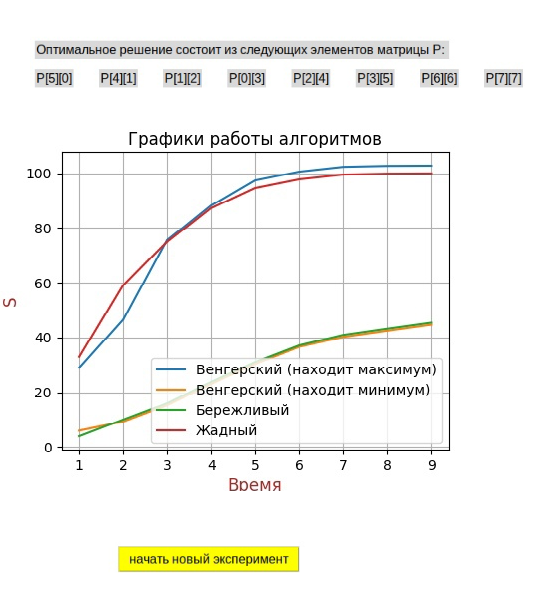


После ввода размерности и нажатия кнопки «Отправить» предлагается ввести нижнюю и верхнюю границу сахаристости.

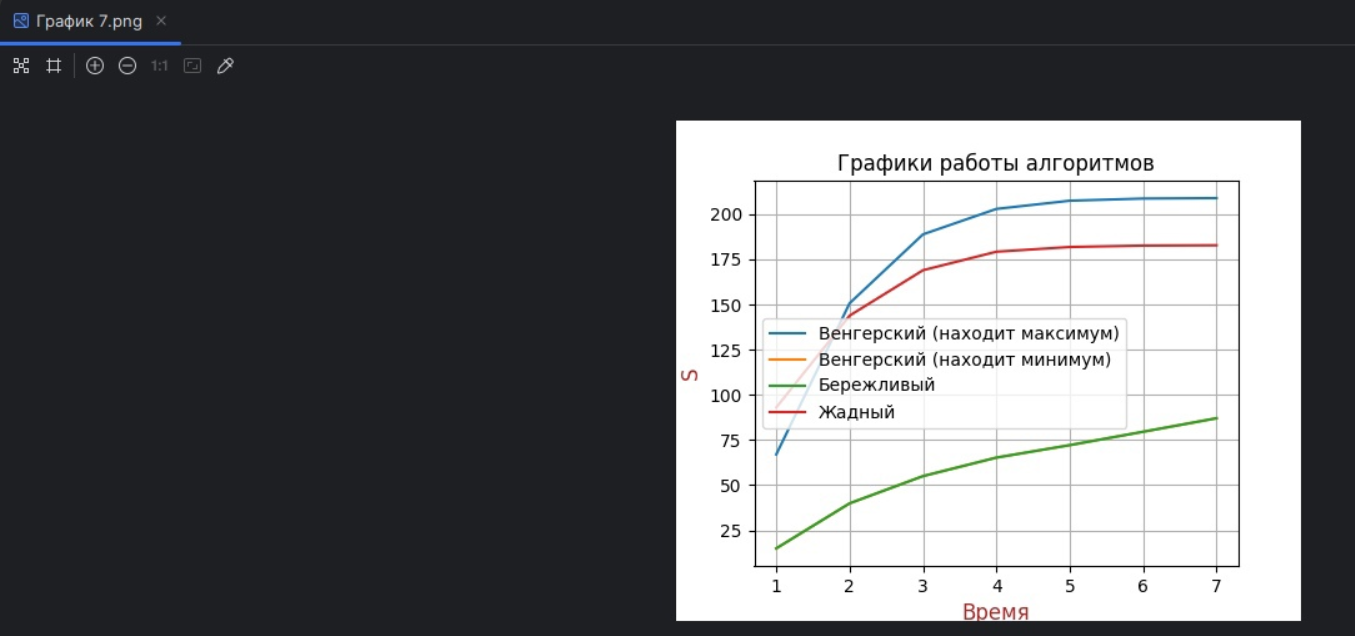
 После ввода необходимых параметров и очередного нажатия кнопки «Отправить» появляется возможность ввести Нижнюю и Верхнюю границу значений деградации.

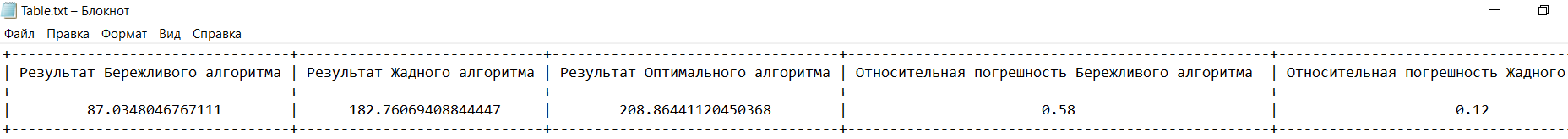
После ввода выше указанных параметров на экран выводится матрица производственной ценности, P (см. на следующей странице)

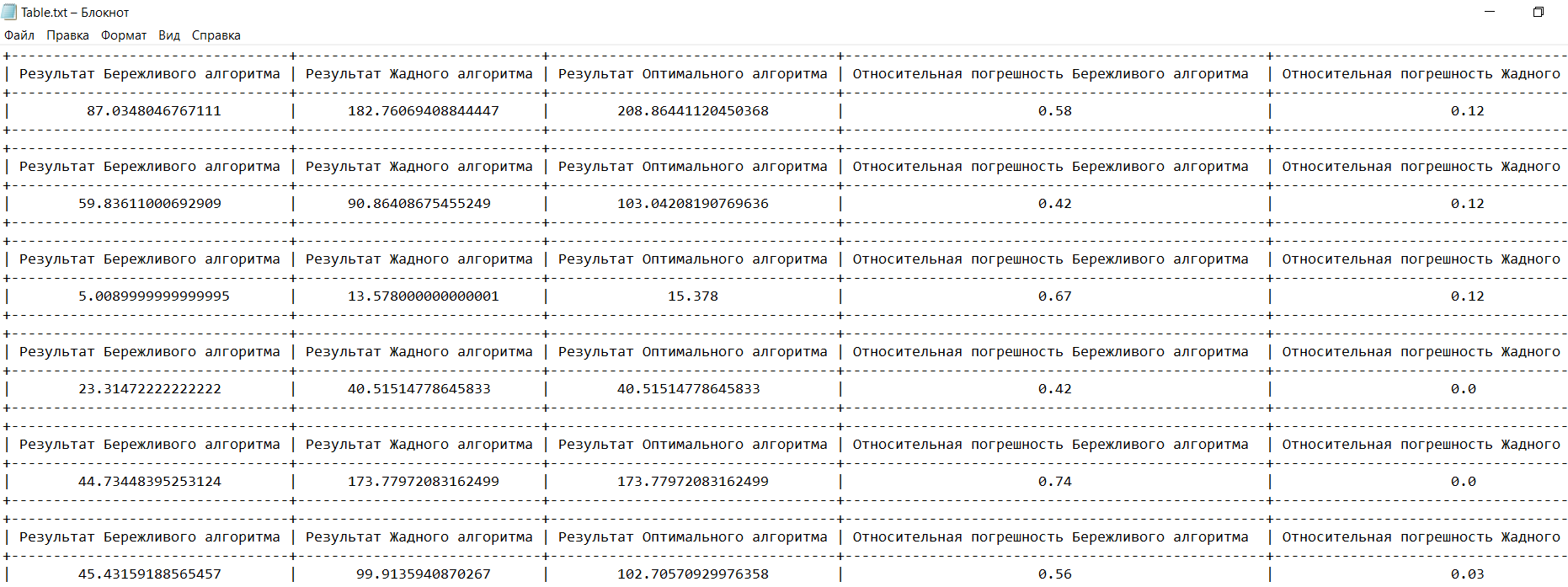
При нажатии кнопки «Построить график» на экране отображается оптимальное решение, а под ним строится график, отображающий работу алгоритмов.



Одновременно с этим в директории приложения сохраняется график в формате .PNG, в той же директории создается файл «Table.txt», куда записываются результаты эксперимента.





При нажатии на кнопку «Новый эксперимент» можно проводить следующий эксперимент, результаты которого сохранятся в тот же файл «Table.txt».

Таким образом можно получить сводку значений относительных погрешностей Жадного и Бережливого алгоритмов и сделать вывод о том, что результаты Жадного алгоритма в ряде случаем оказываются более близкими к значению Оптимального (максимума).

**Дополнительное задание.**

На 2 и [ n/2] этапах происходит поломка оборудования и ремонт

длится сутки. Число дней работы равно n+2.

**Решение:**

Так как число дней простоя на заводе составляет два дня, то нужно сгенерировать еще 2 столбца в матрице коэффициентов деградации. Таким образом, умножаем значения сахаристости а[i][j] в день поломки и в каждый j-й день после на соответствующий ему коэффициент деградации b[i][j].

Число n столбцов матрице сахаристости А - это число дней работы завода, а число столбцов в матрице B - фактическое число дней на обработку заводом n партий.

6. Заключение

Все поставленные задачи были успешно выполнены. Созданная программа обладает простым и понятным графическим интерфейсом. В ходе ряда проведенных экспериментов были получены следующие выводы:

в большинстве экспериментов Венгерский алгоритм оказывался лучше жадного и бережливого алгоритмов, иногда результаты работы жадного и бережливого алгоритмов совпадали с венгерским. В целом, венгерский алгоритм показал себя лучше. Но его вычислительная сложность выше, чем у жадного и бережливого алгоритмов.

7. Список литературы

1. Гагина Е.В., Кузенков О.А. Образовательные стандарты Нижегородского государственного университета им. Н.И.Лобачевского // Вестник Нижегородского университета им. Н.И. Лобачевского. 2014. № 3-4. С. 39-44.

2. Манжесов В.И. Технология хранения, переработки и стандартизации продукции растениеводства. Санкт-Петербург: Троицкий мост, 2010.

3. Харченко С. В. Формирование первичной учетно-аналитической информации в целях учета и контроля при переработке сахарной свеклы на предприятиях сахарной промышленности // Экономика: вчера, сегодня, завтра. 2020. Т. 10. № 2-1. С. 407-419.

4. Кухар В.Н. и др. Методы оценки технологических качеств сахарной свёклы с использованием показателей содержания калия, натрия и α - аминного азота, определённых в свёкле и продуктах её переработки // Сахар. 2019. № 1. С. 18–36.

5. Баландин Д.В. и др. Оптимальный график переработки сахарной свеклы в условиях неопределенности / Сборник трудов Международной научной конференции «Актуальные проблемы прикладной математики, информатики и механики». Воронеж, 13-15 декабря 2021. C. 328–334.

6. Баландин Д.В. и др. Стратегия переработки партий сахарной свеклы при близких параметрах ее увядания / Сборник трудов Второго всероссийского научно-практического семинара «Математическое и компьютерное моделирование и бизнес-анализ в условиях цифровизации экономики». Нижний Новгород, 22 апреля 2022. С. 10–18.

7. Balandin D.V., Egamov A.I., Kuzenkov O.A., Pristavchenko O.V., Vilydanov V.K. Mathematical Modelling and Optimization of Scheduling for Processing Beet in Sugar Production / In book Balandin, D., Barkalov, K., Meyerov, I. (eds) Communications in Computer and Information Science. V. 1750. 2022. P. 227–238. Doi.org/10.1007/978-3-031-24145-1\_19

8. Balandin D.V., Vilydanov V.K., Kuzenkov O.A., Pristavchenko O.V., Egamov A.I. Educational and Research Project “Optimization of the Sugar Beet Processing Schedule” / In book Voevodin V., Sobolev S., Yakobovsky M., Shagaliev R. (eds). Supercomputing. Lecture Notes in Computer Science, book series (LNCS, volume 13708), 2022. 409–422. Doi: 10.1007/978-3-031-22941- 1\_30

9. Банди Б. Основы линейного программирования М.: Радио и связь, 1989. 176 с.

10. Rainer B. Assignment problems. Society for Industrial ad Applied Mathematics. USA, Philadelphia. 2009.

11. Асанов М.О. Дискретная математика: Графы, матроиды, алгоритмы: Учебное пособие. 2-е изд. / М.О. Асанов, В.А Баранский., В.В. Расин. СПб, 2010. 368 c.

12. Рафгарден Т. Совершенный алгоритм. Жадные алгоритмы и динамическое программирование / Т. Рафгарден. СПб.: Питер, 2020. 256 с.

13. Руководство пользователя библиотекой scipy [Электронный ресурс]. URL:https://docs.scipy.org/doc/scipy/reference/generated/scipy.linalg. expm.html (дата обращения: 15.03.2023)  
14. Харрисон М. Как устроен Python. Гид для разработчиков, программистов и интересующихся. СПб.: Питер, 2019. 272 с.

15. Кормен, Т., Лейзерсон, Ч., Ривест, Р., Штайн, К. Глава 16. Жадные алгоритмы // Алгоритмы: построение и анализ = Introduction to Algorithms / Под ред. И. В. Красикова. — 2-е изд. — М.: Вильямс, 2005. — 1296 с. — ISBN 5-8459-0857-4.

16. Рафгарден Т. Совершенный алгоритм. Жадные алгоритмы и  
динамическое программирование / Т. Рафгарден. СПб.: Питер, 2020. 256

17. [**http://cs.mipt.ru/python/lessons/lab1.html**](http://cs.mipt.ru/python/lessons/lab1.html)- Использование matplotlib pyplot для построения графиков функций.

**18**. <https://devpractice.ru/matplotlib-lesson-2-work-with-pyplot/> - **Matplotlib. Урок 2. Работа с инструментом pyplot**

**19.** <https://devpractice.ru/matplotlib-lesson-4-1-viz-linear-chart/> - **Matplotlib. Урок 4.1. Визуализация данных. Линейный график.**

20. <https://www.youtube.com/watch?v=UpR7cOzUBJA&t=7s> – Задача о назначениях.

**21**. <https://habr.com/ru/articles/422009/> - **Венгерский алгоритм, или о том, как математика помогает в распределении назначений**

**22.** <https://e-maxx.ru/algo/assignment_hungary> - **Венгерский алгоритм решения задачи о назначениях**

**23**. [https://qna.habr.com/q/399232](https://vk.com/away.php?utf=1&to=https%3A%2F%2Fqna.habr.com%2Fq%2F399232) - **Как в Python сделать вывод таблицей?**

**24.** <https://docs-python.ru/packages/modul-prettytable-python/pechat-tablichnyh-dannyh-fajl-modul-prettytable/> - **Печать табличных данных в файл, модуль prettytable в Python.**

25. https://habr.com/ru/articles/521894/ - **Формат таблиц в pandas**