

Министерство образования и науки Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»

Кафедра дискретной математики и
информационных технологий

**РАЗРАБОТКА МОДУЛЕЙ «ЗАТРАТЫ НА ПРОИЗВОДСТВО»
ПРИЛОЖЕНИЯ «МОДЕЛЬ ФОРМИРОВАНИЯ ЗЕМЕЛЬНОЙ РЕНТЫ»**
АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студентки 4 курса 421 группы
направления 09.03.01 — Информатика и вычислительная техника
факультета КНиИТ
Блиновой Алины Олеговны

Научный руководитель
старший преподаватель

М. В. Белоконь

Заведующий кафедрой
доцент, к. ф.-м. н.

Л. Б. Тяпаев

Саратов 2023

ВВЕДЕНИЕ

Экономика играет большую роль в развитии современного государства и общества, так как за счёт экономики производится огромное количество благ, которые помогают человеку существовать в этом мире. Например, сельское хозяйство, которое получило своё начало ещё в древние времена, когда люди занимались собирательством и земледелием. Сейчас сельское хозяйство занимает неотъемлемую часть жизни, и его автоматизация позволит быстро и качественно обрабатывать массивы данных о сборах разной сельскохозяйственной продукции. Чтобы правильно обрабатывать эту информацию, нужно создать приложение, которое будет подсчитывать численные данные и выводить их в удобном для пользователя формате.

Таким образом, целью данной бакалаврской работы является создание сельскохозяйственного приложения «Модель формирования земельной ренты».

Чтобы достичь поставленной цели, нужно выполнить следующие задачи:

1. Рассмотреть расчёт модели земельной ренты в аграрной сфере.
2. Изучить и поработать с похожим приложением.
3. Выбрать среду разработки для данной модели.
4. Выбрать эффективные методы решения формул.
5. Реализовать графический интерфейс пользователя.
6. Реализовать работу с Excel-файлом.

Бакалаврская работа состоит из введения, определений, 6 разделов, а именно: «Цифровая оценочная модель», «Земельная рента в аграрной сфере», «Выбор среды разработки и языка программирования», «Программная реализация модели формирования земельной ренты», «Функции расчёта `calculation()` и функции пересчёта `increment()`», «Создание меню», заключения, списка использованных источников и приложения, доступного на диске. Общий объём работы — 48 страниц, из них 48 страниц — основное содержание, включая 26 рисунков, список использованных источников — 20 наименований.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

В первом разделе «Цифровая оценочная модель» описывается, что из себя представляет цифровая оценочная модель, из чего она состоит и как выглядит. Также сделан обзор на приложение аналогичное создаваемому.

Цифровая оценочная модель представляет собой два блока (базовый и рабочий) с набором таблиц: о затратах на производство растениеводческой продукции (исходные данные), основная («базовая» или «расчётная») матрица; корректировка затрат на производство растениеводческой продукции по текущим индексам (с учётом инфляции) цен поставщиков ресурсов и средств производства (расчётная матрица); корректировка затрат на производство растениеводческой продукции (предпочтительные значения) (расчётная матрица).

Таблица представляет собой набор различных значений, которые характеризуют сбор различных сельскохозяйственных культур на угодьях в разные годы. В качестве примера на рисунке 1 показана «Таблица 1.0», являющаяся главной, так как из неё берутся числовые данные, которые задействуются в других таблицах.

Модель базовая (МБ)													Форма № 9-АПК с 2						
Таблица 1.0 - Затраты на производство растениеводческой продукции (исходные данные)																			
(основная (базовая) матрица 1) (вар II, доля переменных затрат с ватые прочих без амортизации 0,6)																			
Наименование показателя	Коды	Площадь, га		Затраты на производство (включая затраты незавершенного производства прошлых лет), отнесенные на себестоимость продукции отчетного года, тыс. руб.															
		посевная	уборочная	в том числе по статье															
Наименование культуры/ вида продукции				всего (пр5+пр6+пр7+пр8+пр9+пр10+пр11+пр12+пр13)	оплата труда с учетом премии на социаль-ные нужды	семена и посадочный материал	в том числе запасные	минеральные удобрения, бактериальные и другие препараты	органические удобрения	средства защиты растений	покупная энергия всего (электричество, тепло, пар)	в т.ч. газ	Прочие виды энергии	электро-энергия	нефте-продукты, лес, каучук, лесопродукты и растительные масла (включая отходы)	затраты на страхование	сохранение основных средств (исходные данные и расчеты на амортизационный период)	прочие затраты	ит. проект затрат: амортизация
1		2	3	4	5	6	7	8	9	10	11а	11б	11в	11	12	13	14	15.1	15.2
Нормативы, коэффициенты				0,6							0,6	116	11в	0,6	0,5	0,1	14	15.1	15.2
Формулы				пр5+пр6+пр7+пр8+пр9+пр10+пр11+пр12+пр13									пр11а+пр11б					пр15.1+пр15.2	пр15.2*
Итого:	92000	2 004 130,6	1 970 665,1	38 429 270	5 260 056	5 299 226	308 240	1 492 386	44 584	2 661 271	782 662	383 282	399 410	372 301	4 981 210	34 838	4 885 100	12 987 936	4 897 430
Земельная и арендованная культуры на зерно и соевые (включая лес)	92100	1 123 227,9	1 078 172,9	19 899 714	2 643 536	2 643 847	206 253	949 547	27 354	1 388 673	149 801	31 122	118 679	116 648	2 806 180	20 144	2 596 169	6 674 462	2 495 612

Рисунок 1 – Таблица 1.0 — «Базовая модель»

Во втором разделе «Земельная рента в аграрной сфере» рассказано о том, что такое земельная рента и какое влияние она оказывает на кадастровую и рыночную оценку земли, также упомянуты особенности кадастровой оценки земли.

В третьем разделе «Выбор среды разработки и языка программирования» обосновывается выбор программных средств, подобранных для создания приложения. В качестве языка программирования был выбран Python. На данный момент он является одним из самых популярных и удобных для про-

граммирования языков, так как подходит для решения широкого круга задач и применяется на всех популярных платформах. Python широко используется в веб-приложениях, разработке программного обеспечения, в машинном обучении и при обработке больших объёмов данных. Также среды разработки данного языка бесплатные и не требуют никаких коммерческих вложений.

Средой разработки для приложения была выбрана PyCharm Community от компании JetBrains. Продукт имеет две версии: PyCharm Community Edition (бесплатная версия) и PyCharm Professional Edition (расширенная версия продукта с дополнительной функциональностью).

Также язык программирования Python был изучен в курсе «Библиотеки машинного и глубокого обучения в ПД» в рамках дополнительного образования, поэтому он удобен в использовании, так как имеются продвинутое знания.

Данный язык программирования не уступает другим, так как его код максимально упрощён, не имеет никаких лишних символов. Также он гибкий и легко расширяемый: может взаимодействовать с другими программными системами.

Также рассмотрены библиотеки, которые помогут в создании приложения. Это библиотеки Tkinter и Openpyxl. Библиотека Tkinter нужна для создания графического интерфейса. Имеет ряд преимуществ: в нём не так много функционала, поэтому он проще в освоении и понимании. Также в приложении упор делается не на интерфейс, а на данные, которые предоставляются пользователям для дальнейшей работы, поэтому Tkinter выигрывает в своей простоте и ненагруженности. К тому же он полностью бесплатен. Библиотека Openpyxl предназначена для работы с файлами Excel.

В четвёртом разделе «Программная реализация модели формирования земельной ренты» описываются основные этапы работы программной реализации модели формирования земельной ренты, установка нужных библиотек, считывание и обработка данных из файла Excel и создание главного окна.

Сначала на вход подаётся файл с расширением .xlsx.

На выходе получится приложение, позволяющее производить различные манипуляции с файлами .xlsx и имеющее определённый набор функций, реализованных для разных таблиц.

Для того, чтобы создать приложение земельной ренты, нужно импортировать библиотеки.

Tkinter — библиотека для создания графического интерфейса.

Scrollbar нужен для создания полосы прокрутки в окне с данными, messagebox — окно с разными оповещениями или ошибками для пользователя.

Ttk позволяет добавлять разные виджеты, например, Button, Entry, Frame, Label, LabelFrame, Menubutton, PanedWindow, Radiobutton и прочие.

Openpyxl позволяет работать с Excel-файлами.

Load_workbook — это класс для создания рабочих листов.

FileDialog нужна для открытия файла из проводника по выбору пользователя.

Datetime — библиотека, которая помогает управлять датой и временем и отображать их в удобном для пользователя виде.

Zipfile — библиотека, которая даёт возможность пользоваться инструментами для создания, чтения, записи, добавления и составления списка ZIP-файлов. Она может обрабатывать ZIP-файлы, использующие расширения ZIP64 — ZIP-файлы размером более 4 гигабайт.

Библиотека Shutil предлагает ряд высокоуровневых операций с файлами и коллекциями файлов. В частности, предоставляются функции, которые поддерживают копирование и удаление файлов.

Далее необходимо считать имеющиеся сведения, выделить оттуда расчётные формулы и подготовить приложение, в котором будут производиться все нужные расчёты.

Для считывания с файлов был создан отдельный метод — create_list(). После этого было создано главное окно приложения. Оно должно представлять собой меню, где пользователь может выбрать, какую таблицу хочет просмотреть.

Чтобы использование функционала Tkinter стало возможным, класс Window необходимо сделать наследуемым от Tkinter. Также нужно инициализировать Window как родительское окно для всех последующих.

Следующим этапом задаются параметры окна, создаются кнопки, выпадающий список и другое наполнение для создания интерфейса. Главное окно показано на рисунке 2.

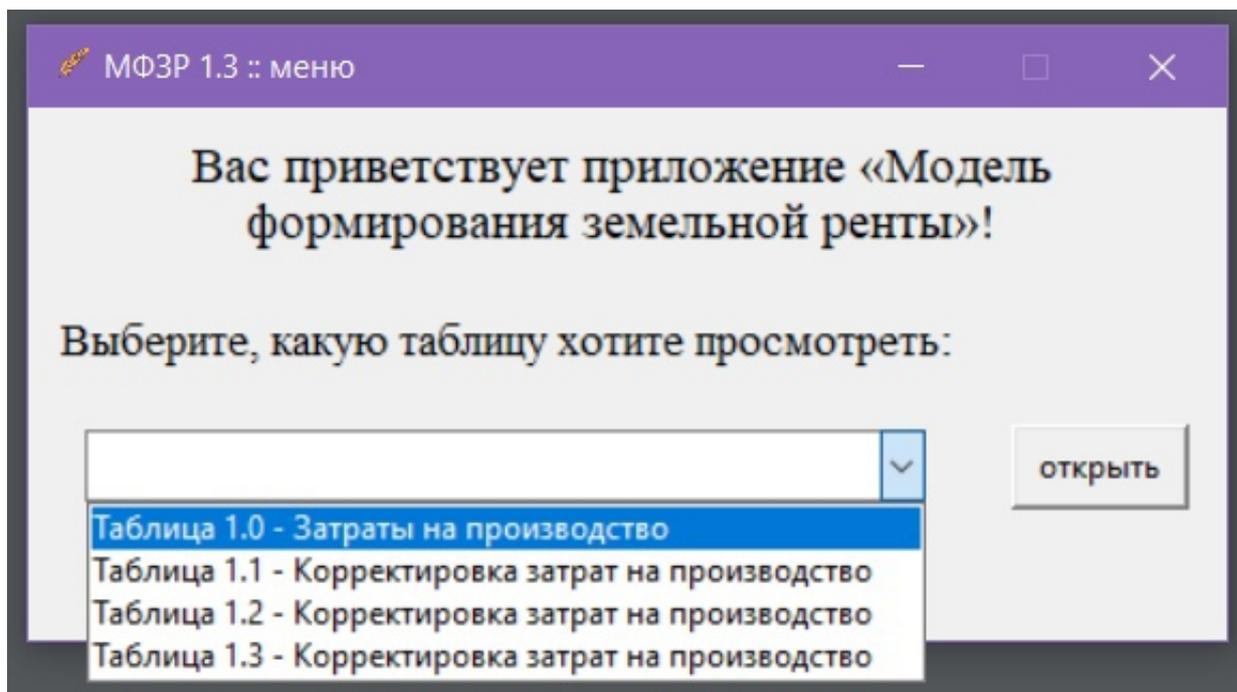


Рисунок 2 – Внешний вид главного окна приложения с раскрытым выпадающим списком

В пятом разделе «Функции расчёта `calculation()` и функции пересчёта `increment()`» описываются методы для произведения расчётов формул в таблицах.

Принцип действия метода `calculation()` заключается в том, что он вызывается в программе в момент обработки считанного массива данных для проведения первоначальных расчётов. В нём осуществляется последовательный проход по всем ячейкам таблицы и определение, находятся ли там формулы. Если ячейка оказалась в нужном диапазоне, производится соответствующий диапазону расчёт.

Метод `increment()` работает похожим образом, но вызывается после каждой вставки нового числа в таблицу. В методе тоже совершается последовательный обход ячеек, аналогичным образом отбираются ячейки, в которых надо произвести расчёты. Затем виджет, отвечающий за отображение таблицы в приложении, очищается и заполняется обновлёнными данными.

На рисунке 3 продемонстрировано изначальное состояние столбцов «прочие затраты», «из прочих затрат: амортизация» и «Прочие без амортизации».

		Семена без элитных	Прочие без амортизации	Всего пер
основных средств (заг материалы, текущий	прочие затраты	из прочих затрат: амортизация		
		15		16
	13.1	13.2	15	15а
			0,6	16
	гр13.1* гр13.1 таб 1.0	гр13.2* гр13.2 таб 1.0	гр6-гр7	гр13.1-гр13.2
				0,6*гр5+гр + гр9+0,6* +0,5*гр12-
	<u>12987935.9921</u>	<u>4897429.6</u>	4990986.6	<u>8090506.392</u>
				17820671.8

Рисунок 3 – Состояние столбцов «прочие затраты», «из прочих затрат: амортизация» и «Прочие без амортизации» до пересчёта

Значение в третьем столбце вычисляется вычитанием второго из первого. Если заменить число 4 897 429.6 на 5000000, в третьем столбце должен получиться ответ 7 987 935,992. И действительно, верный результат показан на рисунке 4.

основных средств (заг материалы, текущий		прочие затраты	из прочих затрат: амортизация	Семена без элитных	Прочие без амортизации	Всего пер
				15		16
	13.1	13.2		15	15а	16
	гр13.1* гр13.1 таб 1.0	гр13.2* гр13.2 таб 1.0		гр6-гр7	0,6 гр13.1-гр13.2	0,6*гр5+гр + гр9+0,6* +0,5*гр12-
	12987935.9921	5000000.0		4990986.6	7987935.992	17759129.6

Рисунок 4 – Состояние столбцов «прочие затраты», «из прочих затрат: амортизация» и «Прочие без амортизации» после пересчёта

Таким образом, расчёт и пересчёт для таблицы 1.0 производится верно. Аналогично и для другой таблицы.

В шестом разделе «Внешний вид приложения» рассказывается об усовершенствовании главного окна приложения и создания меню.

Сначала было создано меню, далее в нём реализовано подменю с помощью переменных, которые добавляются методом `add_cascade`. Также в данном разделе созданы кнопка и событие «Помощь», которые вызывают открытие файла с инструкцией по использованию приложения. Нажатие на клавишу F1 запустит файл помощи для пользователя. Нажатие на клавишу F2 или кнопку «О приложении» провоцирует появление нового окна. Итоговый вариант меню представлен на рисунке 5.

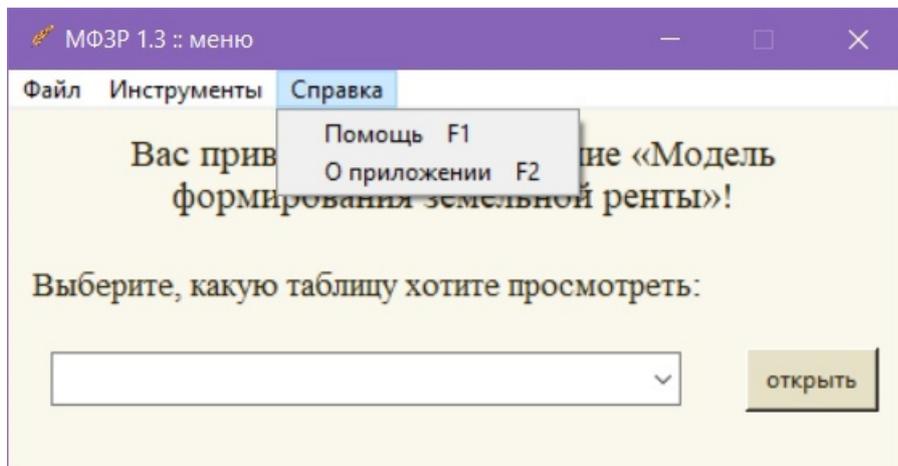


Рисунок 5 – Меню и подменю главного окна приложения

ЗАКЛЮЧЕНИЕ

В ходе выполнения бакалаврской работы было реализовано приложение для оценочной модели формирования земельной ренты. Также была рассмотрена расчётная модель земельной ренты, изучено приложение с похожей функциональностью, реализованы графический интерфейс и работа с файлом формата Excel, применены методы решения формул. Создан пользовательский интерфейс, который помогает людям, использующим приложение, быстро работать с ним, так как он не перегружен большим количеством кнопок и полей. Для разработки приложения был выбран гибкий и расширяемый язык программирования — Python — и соответствующая ему среда разработки — PyCharm Community, подобраны библиотеки для работы и произведен анализ особенностей кадастровой и рыночной стоимости земельного участка.

Таким образом, все задачи работы были выполнены, цель достигнута.

Основные источники информации:

- 1 Земельная рента [Электронный ресурс] URL: <https://www.booksite.ru/fulltext/1/001/008/045/797.htm> (дата обращения: 15.01.2023)
- 2 Особенности оценки земельных участков [Электронный ресурс] URL: <https://www.uphill.ru/press-czentr/stati/osobennosti-otsenki-zemelnyih-uchastkov/> (дата обращения: 30.01.2023)
- 3 Python openpyxl - read, write Excel xlsx files in Python: [Электронный ресурс] URL: <https://zetcode.com/python/openpyxl/> (дата обращения: 13.02.2023)
- 4 tkinter — Python Interface to Tcl/Tk — Python 3.11.4 documentation: [Электронный ресурс] URL: <https://docs.python.org/3/library/tkinter.html> (дата обращения: 21.02.2023)
- 5 John E. Grayson Python and Tkinter Programming // Manning. - 2000 - С. 1-658.
- 6 Г. Россум, Ф. Л. Дж. Дрейк, Д.С. Откидач Язык программирования Python // 2001 - С. 1-454.
- 7 Федоров Д. Ю. Программирование на языке высокого уровня Python // Москва : Издательство Юрайт - 2019. - С. 1-161.