Министерство науки и высшего образования РФ

ФГАОУ ВО «Уральский федеральный университет   
имени первого Президента России Б.Н. Ельцина»

Институт новых материалов и металлургии

Кафедра «Теплофизика и информатика в металлургии»

Оценка работы: \_\_\_\_\_\_\_\_\_\_\_\_\_

Члены комиссии:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись расшифровка подписи

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись расшифровка подписи

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2021 г.

Расчет количества оксидов азота, образующихся в рабочем пространстве пламенных печей

**ПРОЕКТ ПО МОДУЛЮ**

**«Металлургические технологии»**

Направление 09.03.02 «Информационные системы и технологии»   
(уровень бакалавриата)

Образовательная программа 09.03.02/03.01   
«Информационные системы и технологии в металлургии»

Руководитель

доцент, к.т.н. И.А. Гурин

должность, звание подпись расшифровка подписи

Консультант

доцент, к.т.н. В.А. Гольцев

должность, звание подпись расшифровка подписи

Консультант

профессор, д.т.н. В.В. Лавров

должность, звание подпись расшифровка подписи

Студент:

НМТ-363907 [А.А. Гафаров]

номер группы подпись расшифровка подписи

Екатеринбург

2021

**Задание**

**на выполнение проекта по модулю «Металлургические технологии»**

Студент группы НМТ-383907. Образовательная программа: 09.03.02/03.01 – «Информационные системы и технологии в металлургии».

Фамилия: Гафаров. Имя: Абдужамил Отчество: Абдужалилович.

Руководитель: доцент Гурин Иван Александрович.

Срок выполнения работы: с «14» сентября 2020 г. по «30» декабря 2020 г.

1. Тема проекта:

Расчет количества оксидов азота, образующих в рабочем пространстве пламенных печей

2. Содержание работы

2.1. Пояснительная записка:

* титульный лист;
* бланк задания преподавателя на выполнение проекта;
* оглавление;
* постановка задачи, проверка корректности алгоритма расчета:
  + физическая постановка задачи;
  + математическая модель (подробно в цифрах);
  + создание тестового варианта расчета в Microsoft Excel;
  + постановки задач для выполнения отдельных этапов проекта;
* проектирование и реализация программного обеспечения:
  + создание архитектуры системы;
  + разработка блок-схемы работы пользователя с программой;
  + создание программного обеспечения в системе управления версиями Atlassian Bitbucket;
  + разработка математической библиотеки;
  + разработка пользовательского веб-интерфейса;
  + обработка исключительных ситуаций;
  + разработка контекстно-зависимой справки;
  + создание дистрибутива;
* описание программного обеспечения:
  + установка и настройка программного продукта;
  + функциональные возможности программного продукта;
  + технология выполнения расчетов на конкретном примере;
* заключение
* библиографический список;
* приложение:
  + фрагменты листинга программного обеспечения.

2.2. Компьютерные версии

* исходный код проекта расположен в системе удаленного контроля версий Bitbucket;
* реализация алгоритма расчета в электронных таблицах Microsoft Excel;
* архитектура программного обеспечения в пакете в пакете Microsoft Visio;
* руководство пользователя в форматах \*.hlp, \*.chm, \*.pdf.

3. Особые дополнительные требования

* программная платформа – операционная система Microsoft Windows;
* среда программирования – Microsoft Visual Studio 2017 (2019). Язык программирования C#;
* объектно-ориентированная методология разработки;
* расчетный модуль в виде библиотеки dll, которая подключена к клиентскому модулю;
* обеспечение функциональности программного обеспечения: ввод/корректировка/сохранение варианта исходных данных; расчет; отображение на форме результатов расчета в численном и графическом виде; формирование отчета с возможностью настройки показателей; файл справочной помощи;
* надежность работы, защита от некорректно вводимых данных;
* установка программного средства на веб-сервер;
* нумерацию версий программы проводить в формате 1.YY.MM.NN, где YY – две последние цифры года, MM – номер месяца, NN – номер версии в текущем месяце.

4. К защите предоставляются следующие материалы

* программная реализация (дистрибутив и исходники программного продукта);
* размещение исходного кода проекта в системе Atlassian Bitbucket; проверка рабочей версии программного продукта путем компиляции из системы удаленного контроля версий Bitbucket в присутствии Заказчика;
* файл в электронных таблицах с тестовой версией алгоритма расчета;
* презентация работы;
* пояснительная записка (ПЗ), выполненная в соответствии с требованиями по оформлению курсовых и дипломных работ.

Допуск к защите проекта по модулю осуществляется только после предварительной сдачи преподавателю всего набора содержимого электронных папок и утвержденной пояснительной записки.

5. План выполнения проекта по модулю:

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование элементов проектной работы | Сроки | Примечания | Отметки о выполнении |
| Анализ предметной области. Проверка корректности алгоритма расчета. Выполнение тестовых расчетов в виде xls-файла | 14.09.2020 –  08.10.2020 гг. | Алгоритм расчета в виде Microsoft Excel |  |
| Создание архитектуры. Проектирование структуры класса математической библиотеки. Реализация алгоритмического обеспечения. Кодирование модуля dll и клиентского модуля. | 09.10.2020 –  29.10.2020 гг. | Пакет Microsoft Visio. Программное обеспечение модуля dll и клиентского модуля |  |
| Подготовка программных тестов для тестирования математической библиотеки | 30.10.2020 –  05.11.2020 гг. | Проект тестирования в Microsoft Visual Studio 2017 (2019) |  |
| Совершенствование клиентского модуля: отображение в программе результатов расчета в численном виде | 06.11.2020 –  12.11.2020 гг. | Visual Studio 2017 (2019) |  |
| Совершенствование клиентского модуля: отображение в программе результатов расчета в графическом виде | 13.11.2020 –  19.11.2020 гг. | Visual Studio 2017 (2019) |  |
| Совершенствование клиентского приложения: формирование отчета в программе | 20.11.2020 –  26.11.2020 гг. | Visual Studio 2017 (2019) |  |
| Совершенствование клиентского приложения: разработка справочной помощи | 27.11.2020 –  03.12.2020 гг. |  |  |
| Оформление пояснительной записки. Подготовка презентации. | 04.12.2020 –  30.12.2020 гг. | Сдача всего содержимого электронных папок и ПЗ на проверку |  |

6. Выполнение проекта по модулю закончено «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2020 г.

Руководитель работы, доцент, к.т.н. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ И.А. Гурин

Задание к исполнению принял(а) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.А. Гафаров

**РЕФЕРАТ**

Пояснительная записка изложена на 29 листах и содержит 4 таблицу, 13 рисунков и 1 приложение.

РАСЧЕТ КЕРАМИЧЕСКИХ РЕКУПЕРАТОРОВ, ПРОГРАММИРОВАНИЕ, ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, ИНТЕРФЕЙС, VISUAL STUDIO C#, ИНСТАЛЛЯЦИЯ.

Проект по модулю посвящен разработке программного обеспечения для расчета керамических рекуператоров.

Отражены основные этапы разработки программного обеспечения: постановка задачи, реализация тестового варианта расчета в электронных таблицах Microsoft Excel; проектирование и реализация программного средства – математической библиотеки и пользовательского интерфейса; разработка системы автоматизированного тестирования математической библиотеки; создание справочной документации; подготовка дистрибутива. Размещение исходного программного кода выполнено в системе удаленного контроля версий Bitbucket.

Основными функциями программного обеспечения является расчет керамических рекуператора определения поверхности нагрева, которая должна обеспечить нагрев требуемого для горения количества воздуха до заданной температуры.

Основные конечные пользователи программного обеспечения – студенты вузов.

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 6](#_Toc61815404)

[1 ПОСТАНОВКА ЗАДАЧИ, ПРОВЕРКА КОРРЕКТНОСТИ АЛГОРИТМ РАСЧЕТА 7](#_Toc61815405)

[1.1 Физическая постановка задачи 7](#_Toc61815406)

[1.2 Математическая модель задачи 7](#_Toc61815407)

[1.3 Создание тестового варианта расчета в электронных таблицах Microsoft Excel 12](#_Toc61815408)

[2 ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ 14](#_Toc61815409)

[2.1 Разработка архитектуры системы 14](#_Toc61815410)

[2.2 Разработка блок-схемы работы пользователя с программой 15](#_Toc61815411)

[2.3 Создание программного обеспечения в системе управления версиями Atlassian Bitbucket 16](#_Toc61815412)

[2.4 Разработка математической библиотеки 17](#_Toc61815413)

[2.5 Реализация пользовательского интерфейса 18](#_Toc61815414)

[2.6 Обработка исключительных ситуаций 19](#_Toc61815415)

[2.7 Создание справочной помощи 20](#_Toc61815416)

[2.8 Создание дистрибутива 20](#_Toc61815417)

[3 ОПИСАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ 21](#_Toc61815418)

[3.1 Установка и настройка программного средства 21](#_Toc61815419)

[4.2 Функциональные возможности программного продукта 22](#_Toc61815420)

[4.3 Технология выполнения расчетов 23](#_Toc61815421)

[ЗАКЛЮЧЕНИЕ 24](#_Toc61815422)

[БИБЛИОГРАФИЧЕСКИЙ СПИСОК 25](#_Toc61815423)

[ПРИЛОЖЕНИЕ 26](#_Toc61815424)

# ВВЕДЕНИЕ

В современных пламенных печах, предназначенных для осуществления различных высокотемпературных процессов, необходимая технологическая температура достигается за счет химической энергии топлива, выделяемой в результате его горения.

Теплота, выделяемая при сжигании топлива в печах, только частично расходуется на технологический материал и восполнение тепловых потерь рабочим пространством печи в окружающее пространство. Большая часть теплоты (иногда до 80 %) требуется продуктами горения, покидающими рабочее пространство печи. Поэтому вопрос утилизации теплоты уходящих из печи газов приобретает первостепенное значение. Для этого используют теплообменные аппараты непрерывного и периодического действия. Очень выгодно использовать теплоту уходящих газов для подогрева воздуха, поступающего для горения топлива. В этом случае каждая единица теплоты (кДж), внесенная в печь с подогретым воздухом, позволяет экономит 2-3 и более единицы теплоты топлива (в зависимости от уровня подогрева воздуха и температуры уходящих газов). Это обуславливается тем, что внесение в печь и физической теплоты нагретого воздуха или газообразного топлива уменьшает расход последнего и, следовательно, объем продуктов сгорания, что приводить к снижению тепловых потерь.

Подогрев воздуха помимо экономии топлива интенсифицирует процесс горения газообразного топлива так как с повышением температуры газовоздушной смеси возрастают скорости реакции горения и распространение пламени.

Для подогрева воздуха в настоящее время применяют аппараты как непрерывного действия – рекуператоры, так и периодического – регенераторы. Последними оборудуют доменные печи, высокотемпературные плавильные печи и регенеративные нагревательные колодцы. Наиболее высокий уровень подогрева воздуха можно получить в регенераторах (до 900-1300 ˚С) и керамических рекуператорах (до 700-850 ˚С). Керамические рекуператоры компонуются из специальных элементов – блоков или труб, поэтому и носят название либо блочные, либо трубчатых.

# 1 ПОСТАНОВКА ЗАДАЧИ, ПРОВЕРКА КОРРЕКТНОСТИ АЛГОРИТМ РАСЧЕТА

## 1.1 Физическая постановка задачи

Целью является расчет керамического рекуператора.

Керамический рекуператор зависит от:

* Расхода топлива на печи.
* Коэффициента избыточного воздуха.
* Характеристики горения топлива.
* Температуры продукта горения, уходящих из печи.

## 1.2 Математическая модель задачи

Исходными формулами для расчета рекуператора являются следующие выражения.

Уравнение теплопередачи, решенное относительно искомой поверхности нагрева,

(1)

где F –требуемая поверхность нагрева рекуператора, м²;

–количества теплоты, передаваемое воздуху в рекуператоре, %;

K –коэффициент теплопередачи в рекуператоре, Вт/ м²град;

– средняя логарифмическая разность температур дыма и воздуха в рекуператоре, град.

Уравнения теплового баланса, записанное с учетом утечки воздуха на дымовую сторону вследствие негерметичности рекуператора:

(2)

где - коэффициент, учитывающий потери топливо через наружные поверхности рекуператора в окружающее пространство (обычно );

– количество дыма, поступающее в рекуператор, м³·с;

– количество воздуха, поступающее в рекуператор, м³·с;

– количество воздуха, просасываемое через неплотности на сторону дыма в рекуператоре, м³·с;

– теплосодержание дыма до и после рекуператора, Дж/ м³;

– теплосодержание воздуха до и после рекуператора, Дж/ м³;

Из уравнения (2) получаем теплосодержание дыма, покидающего рекуператора;

(3)

что дает возможность определить температуру дыма .

Для прямоточного движения дыма и воздуха в рекуператоре среднюю логарифмическую разность температур рассчитывают по формуле

(4)

Для противоточного движения формула имеет вид сгорания:

при сжигании топлива в смеси с атмосферным воздухом

(5)

Для более сложных схем движения (перекрестного – прямоточной или перекрестно – противоточной) в формуле (4) или (5) вводят поправочный коэффициент , для нахождения которого вычисляют вспомогательные величины:

(6)

(7)

Зная R и P, по графикам (прило.1) находят поправочный коэффициент , на которых умножают результат, полученной по формуле (4) и (5).

Коэффициент теплопередачи K определяют по формуле

(8)

где – коэффициент теплоотдачи от дыма к стенке рекуператора, Вт/ м²град

– коэффициент теплоотдачи от стенки к воздуху, Вт/ м²град:

– толщина стенки рекуператора, м²;

– коэффициент теплопроводности материала стенки рекуператора, Вт/ м²град.

Коэффициент теплоотдачи на дымовой стороне равен

(9)

где коэффициент теплоотдачи конвекцией, Вт/ м²град;

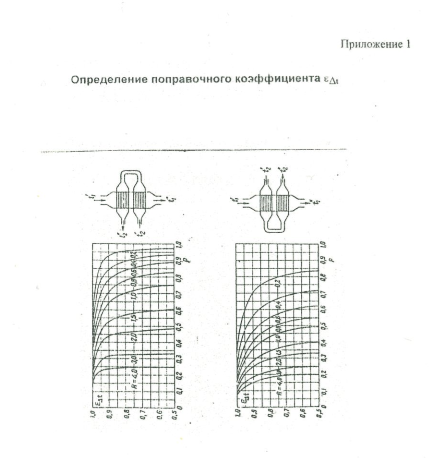
коэффициент теплоотдачи излучение, Вт/ м²град;

На воздушную стороне учитывают только теплоотдачу конвекцией.

Способы расчета зависит от конструкции рекуператора и характера движения дыма и воздуха и будет показаны для каждого конкретного случая в примерах расчета.

Приложение 1

**Определения поправочного коэффициента**



Приложение 2

**Энтальпия продуктов сгорания топлива**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| t,°C | ,кДж/м³ | | | t,°C | ,кДж/м³ | | |
| **I** | **II** | **III** | **I** | **II** | **III** |
| 100 | 134 | 146 | 148 | 1300 | 2066 | 2090 | 2135 |
| 200 | 283 | 292 | 294 | 1400 | 2242 | 2267 | 2327 |
| 300 | 432 | 440 | 442 | 1500 | 2417 | 2445 | 2508 |
| 400 | 581 | 588 | 591 | 1600 | 2596 | 2626 | 2690 |
| 500 | 730 | 736 | 743 | 1700 | 2780 | 2806 | 2880 |
| 600 | 879 | 890 | 908 | 1800 | 2968 | 2990 | 3072 |
| 700 | 1028 | 1046 | 1066 | 1900 | 3158 | 3177 | 3266 |
| 800 | 1197 | 1220 | 1236 | 2000 | 3346 | 3365 | 3460 |
| 900 | 1369 | 1392 | 1411 | 2100 | 3536 | 3536 | 3660 |
| 1000 | 1542 | 1565 | 1591 | 2200 | 3727 | 3727 | 3858 |
| 1100 | 1714 | 1739 | 1770 | 2300 | 3916 | 2935 | 4048 |
| 1200 | 1890 | 1914 | 1952 | 2400 | 1407 | 4127 | 4260 |
|  |  |  |  | 2500 | 4299 | 4329 | 4464 |

Примечания

1. **I** группа топлив: природных и коксовый газы, смесь последнего доменным приР>12000 кДж/м³;

**II** группа топлив: смесь коксового и доменного газов при Р=8000…12000 кДж/м³, мазут, кокс, каменный уголь;

**III** группа топлив: смесь коксового и доменного газов при Р<8000 кДж/м³, доменный газ, торф, бурый уголь.

2.В найденную по таблице температуру продуктов сгорания необходимо внести поправку, учитывающую влияние избыточного воздуха на температуру. Для этого процентное содержание воздуха в продуктах сгорания следует умножить на поправку и полученное значение прибавить к найденной по таблице температуре. Величина берется в зависимости от интервала температуры:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | °C | <1200 | 1300-1500 | 1600-1700 | 1800-2000 | 2100-2200 | 2300-2500 |
|  | °C | 1 | 2 | 3 | 4 | 5 | 6 |

Приложение 3

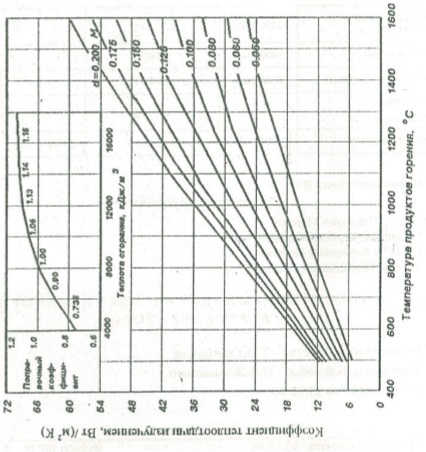
**Средняя теплоемкость газообразных видов топлива и воздуха**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| t,°C | , кДж/м³ | | | | t,°C | , кДж/м³ | | | |
| **п** | **к** | **д** | **в** | **п** | **к** | **д** | **в** |
| 0 | 1,55 | 1,35 | 1,33 | 1,29 | 700 | 2,36 | 1,63 | 1,43 | 1,36 |
| 100 | 1,64 | 1,39 | 1,34 | 1,29 | 800 | 2,45 | 1,66 | 1,45 | 1,38 |
| 200 | 1,76 | 1,43 | 1,35 | 1,31 | 900 | 2,56 | 1,70 | 1,46 | 1,39 |
| 300 | 1,90 | 1,47 | 1,37 | 1,32 | 100 | 2,66 | 1,73 | 1,48 | 1,40 |
| 400 | 2,02 | 1,51 | 1,39 | 1,33 | 1100 | - | 1,76 | 1,49 | 1,41 |
| 500 | 2,14 | 1,55 | 1,41 | 1,34 | 1200 | - | 179 | 1,50 | 1,42 |
| 600 | 2,27 | 1,59 | 1,42 | 1,35 | 1300 | - | 1,81 | 1,51 | 1,43 |

Примечание: п-природный, к-коксовый, д-доменный, в-воздух.

Приложения 4

**Коэффициент теплоотдачи излучением от дыма к стенкам канала**



Приложения 5

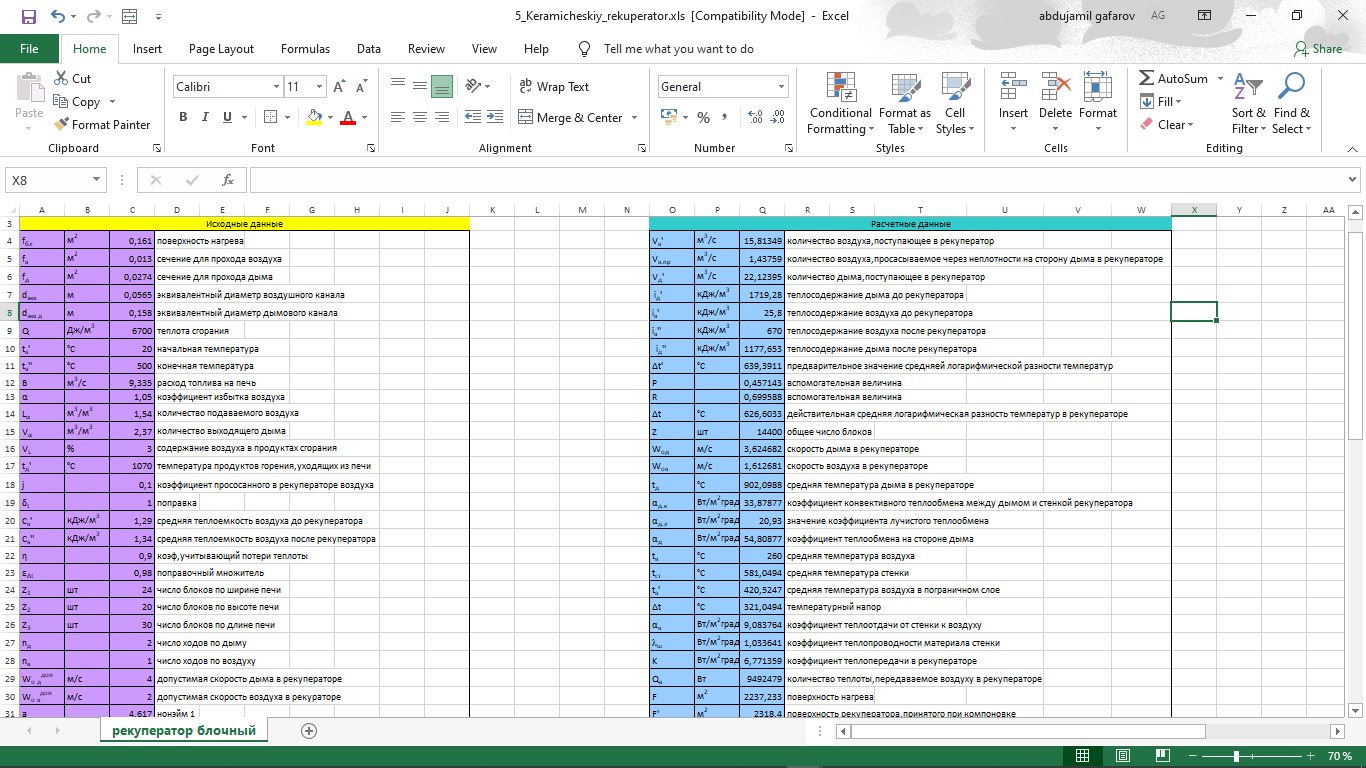
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Материалы | Плоность,  кг/м³ | Коэффициент  Теплопроводности,  Вт/м·град | Средняя  теплоемкость,  кДж/кг·град | Придельная  Температура  применения |
| Иносовы обычное | 1900-2000 | 0,82+6,8 В | 0,87+1,9 В | 1700 |
| Шамотные | 1800-2000 | 0,7+6,4 В | 0,87+2,1 В | 1350 |
| Шамотные класса А | - | 0,9+2,3 В | 0,87+2,1 В | 1400 |
| Форстеритовые  Насадочные | 2350-2500 | 4,23-16 В | - | 1400 |
| Магнезитовые | 2600-2800 | 6,3-27 В | - | 1700 |
| Хромомагнезитовые | 2700-2850 | 2,8-8,7 В | - | 1700 |
| Магнезито-  хромитовые | - | 4,1-16 В | 1 | 1750 |
| Карбошамотные | 1900-2000 | 0,93+8,5 В | 0,87+2,1 В | 1400 |

Примечания: В=1 t.

## 1.3 Создание тестового варианта расчета в электронных таблицах Microsoft Excel

После ознакомления с методическим пособием, был формализован алгоритм расчета в электронных таблицах Microsoft Excel.

Входные данные для расчета блочного керамического показано на рисунке (1.3.1).



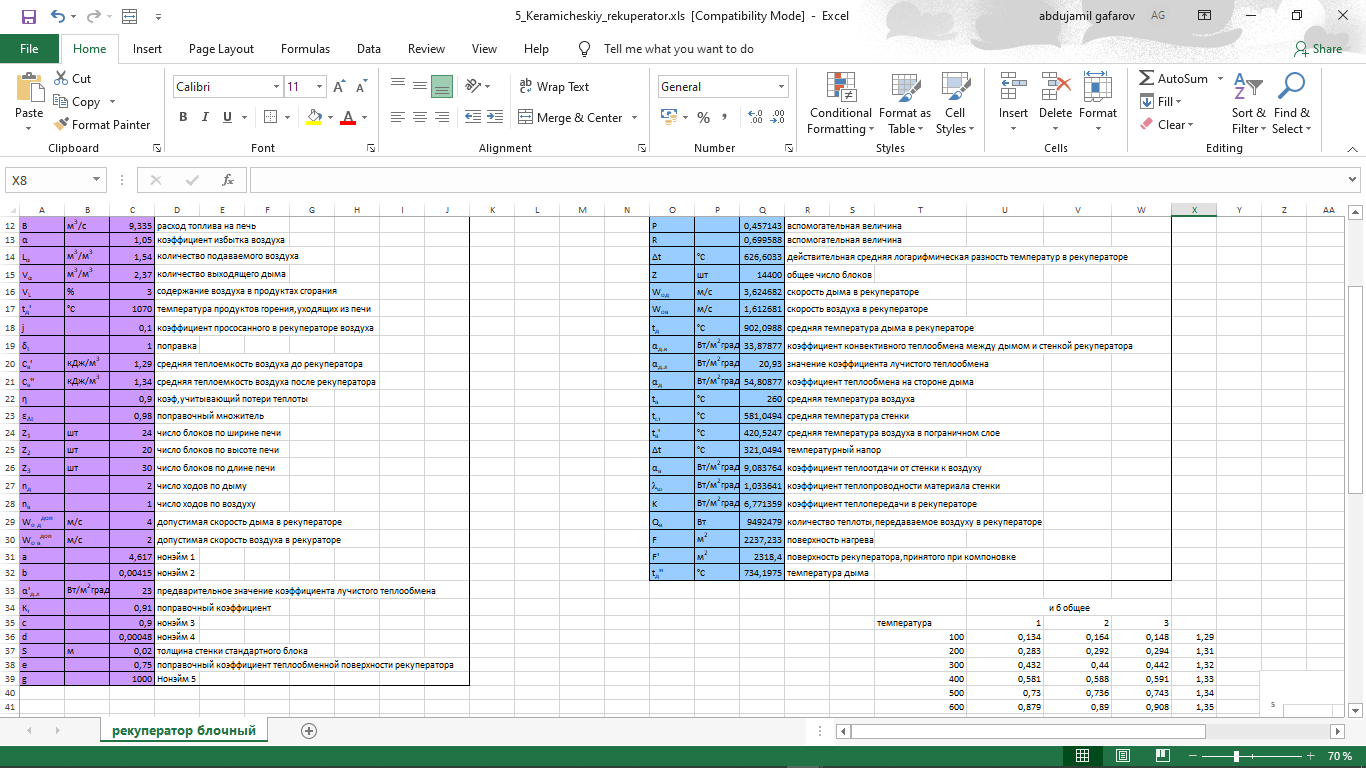
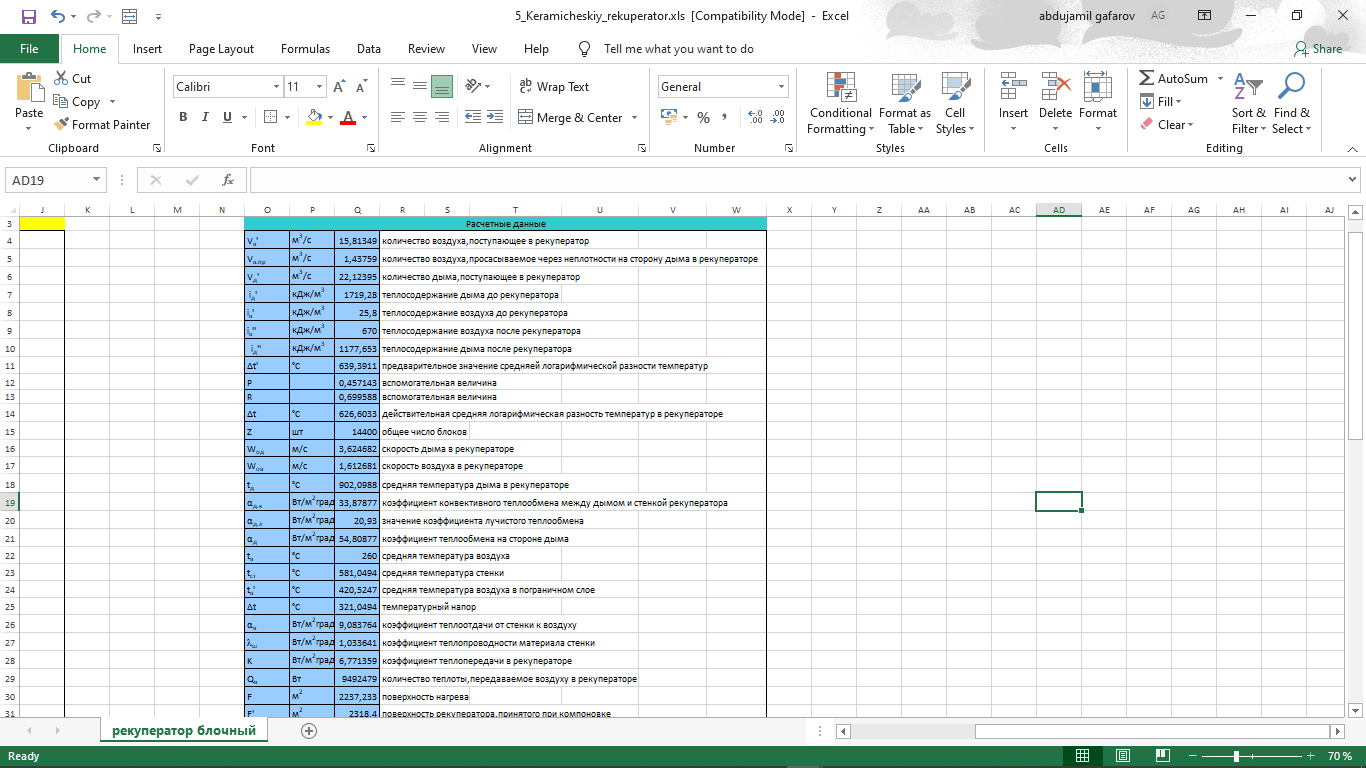


Рисунок 1.3.1- Входные данные

Все расчеты были сделаны в соответствии с формулами, которые приведены в методическом пособии показано на рисунке (1.3.2).



# 2 ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

## 2.1 Разработка архитектуры системы

В процессе решения поставленных задач нужно спроектировать архитектуру информационной системы. Создав файл Microsoft Office Excel 2017, мы перешли к разработке программного продукта. Составили примерную модель, в которой отобразили все требуемые функции заказчика.

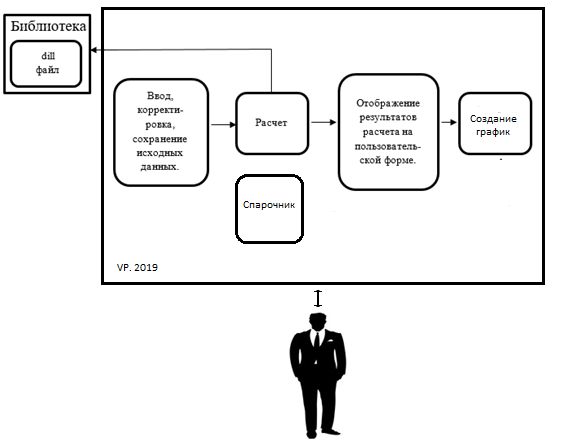


Рисунок 2.1.1– Разработки архитектуры системы

## 2.2 Разработка блок-схемы работы пользователя с программой

Блок-схема-распространенный тип схем(графических моделей), описывающих алгоритмы или процессы, в которых отдельные шаги изображаются в виде блоков различной формы, соединенных между собой линиями, указывающими направление последовательности.

Пользователь запускает программу. Программа предоставляет возможность выбрать задачи решения. После ввода исходных данных, пользователь может рассчитать нужные величины. Затем ему выдается результат расчетов. После пользователь может завершить работу с программой.

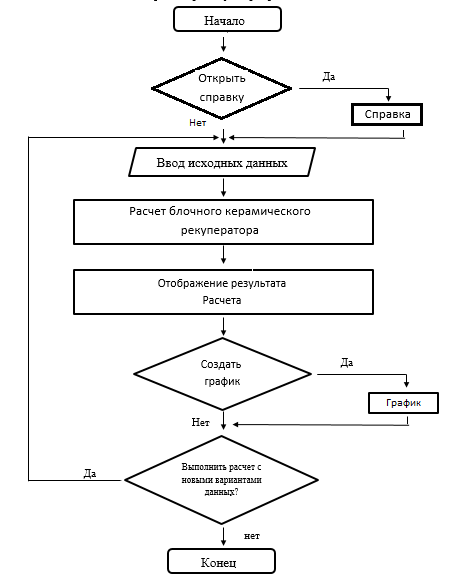


Рисунок 2.2.2 – Блок-схема работы пользователя с программой

## 2.3 Создание программного обеспечения в системе управления версиями Atlassian Bitbucket

Bitbucket — веб-сервис для хостинга проектов и их совместной разработки, основанный на системе контроля версий Mercurial и Git. По назначению и предлагаемым функциям аналогичен GitHub (однако GitHub не предоставляет бесплатные «закрытые» репозитории, в отличие от Bitbucket), который поддерживает Git и Subversion.

Bitbucket — это неограниченное количество закрытых и открытых репозиториев. Команды до 5 человек могут использовать Bitbucket Cloud бесплатно.

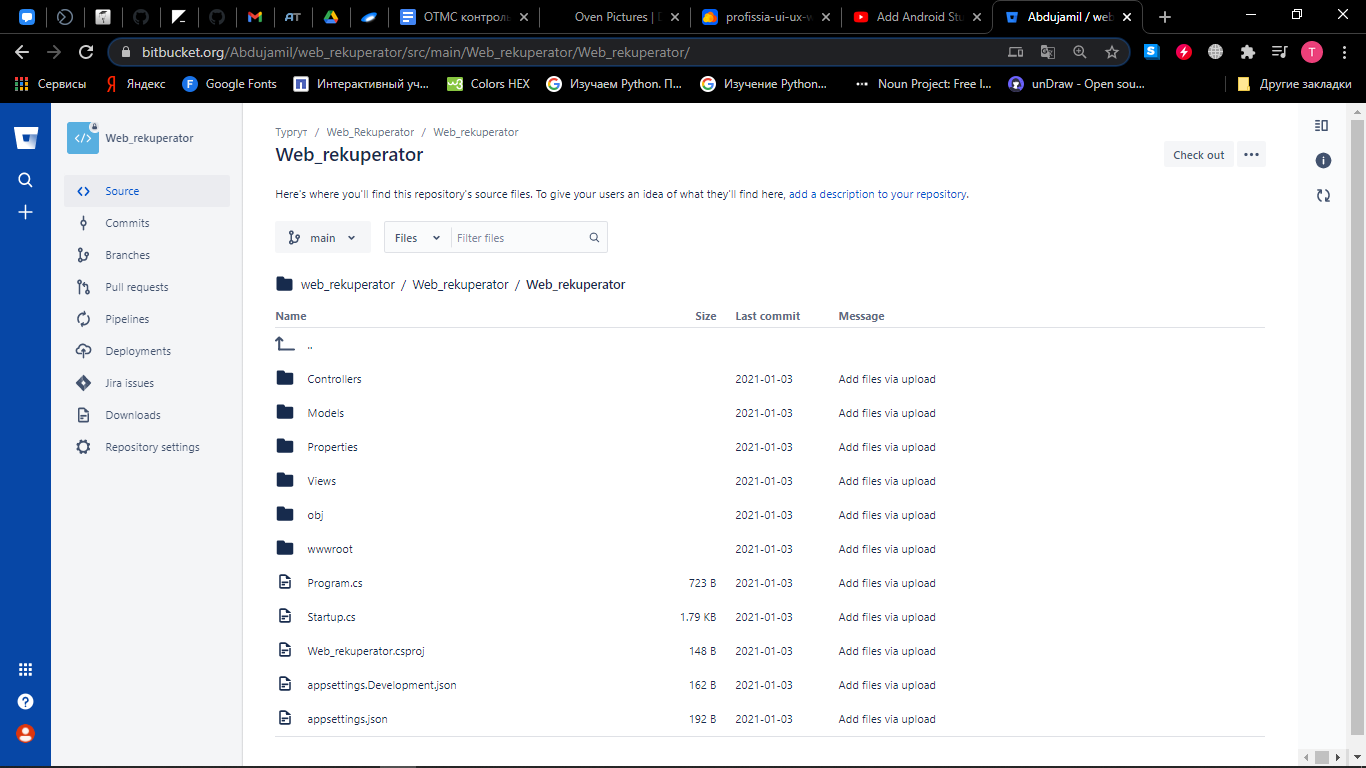


Рисунок 2.3.1 – Репозиторий в Bitbucket

## 2.4 Разработка математической библиотеки

Библиотека реализованна в среде Visual Studio в виде dll-файла. Алгоритм расчета реализован на основе Excel-файла, DFD-диаграмм и спецификации. Наша библиотека специфическая и узко специализированная.

Фрагмент среды разработки в коде математической библиотеки представлен на рисунке 2.4.1.

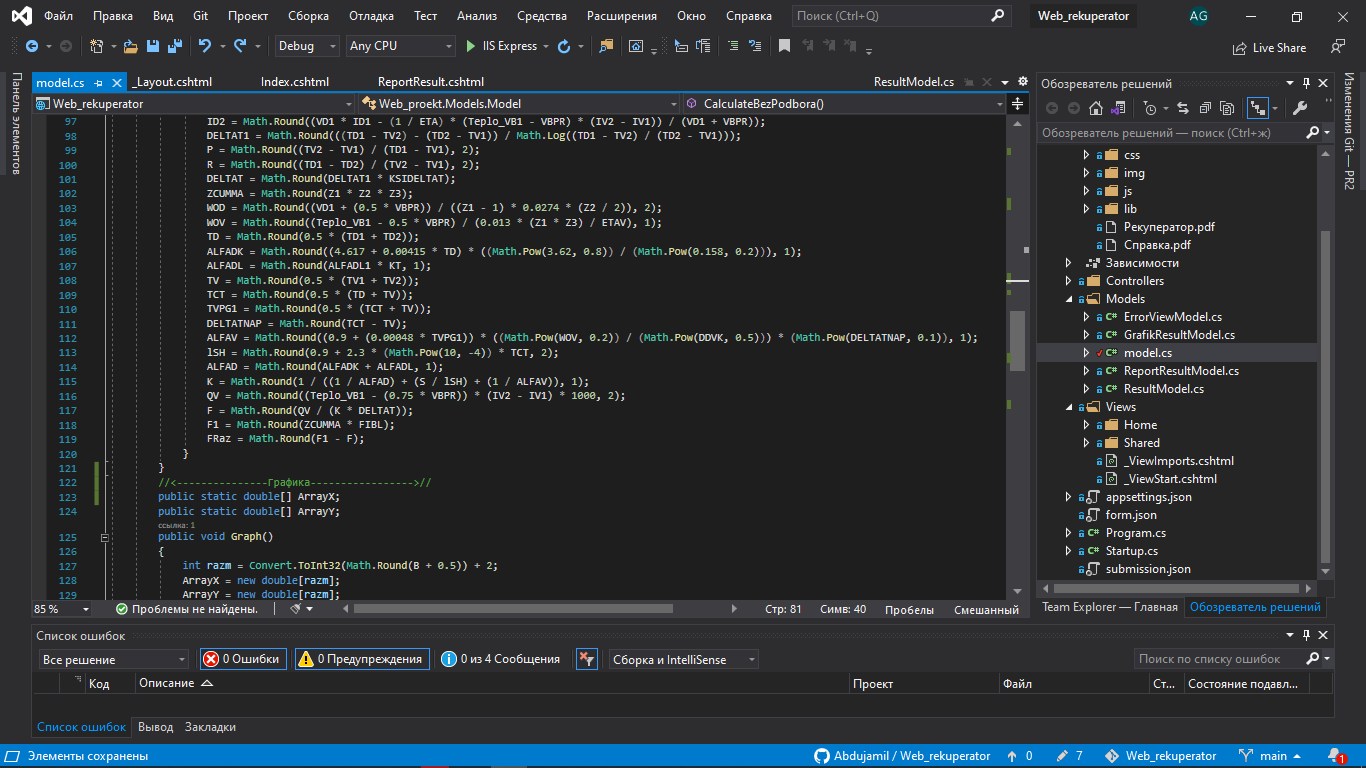


Рисунок 2.4.1 – Фрагмент среды разработки в коде математической

библиотеки

## 2.5 Реализация пользовательского интерфейса

Интерфейс пользователя, он же пользовательский интерфейс – интерфейс, обеспечивающий передачу информации между пользователем-человеком и программно-аппаратными компонентами компьютерной системы.

Пользовательский интерфейс включает в себя следующие пункты:

* Рекуператор

При нажатии на «Рекуператор» ссылается на веб-страницу, где показывается информация о Рекуператоре.

* Кнопка «Вычислить результат»

При нажатии на кнопку «Вычислить результат» программа произведет вычисления, результат которых будет во вкладке «Результат расчета».

* Кнопка «Очистить таблицу»

При нажатии на кнопку «Очистить таблицу» программа очищает значения в таблице «Исходные данные».

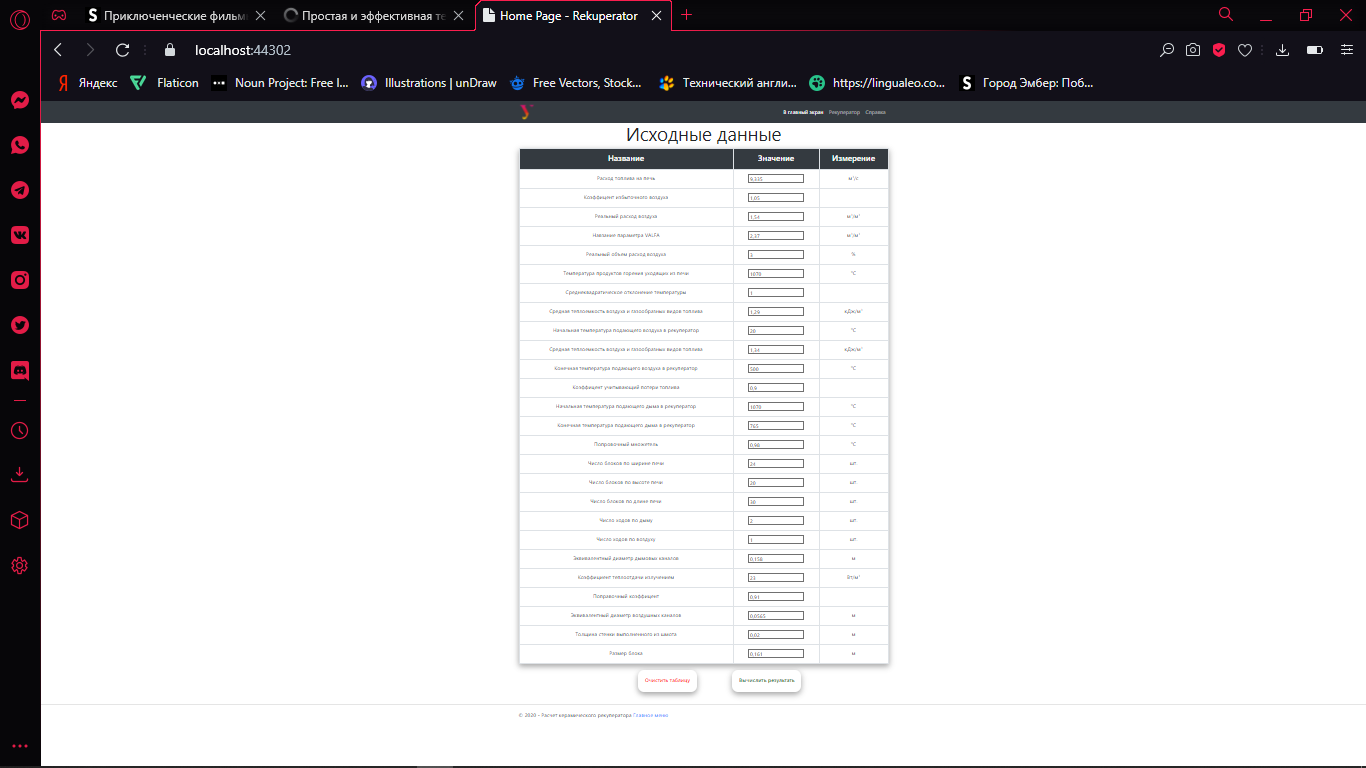
* Кнопка «Построит график»

При нажатии на кнопку «Построит график» откроется вкладка «График», на котором будет построена График зависимости Количество продуктов горения перед рекуператором от Расхода топлива на печь.

* Справочник

При нажатии на «справочник» откроется документ, содержащий справочную информацию и описание работы с программой.

Фрагмент окна пользовательского интерфейса в среде Visual Studio представлен на рисунке 2.6.1.



## 2.6 Обработка исключительных ситуаций

## 2.7 Создание справочной помощи

Методические пособия сообщают нам сведения, побуждающие принимать определенные решения, т.е. инициируют управленческие решения, позволяют выбрать тот или иной способ управленческого воздействия.

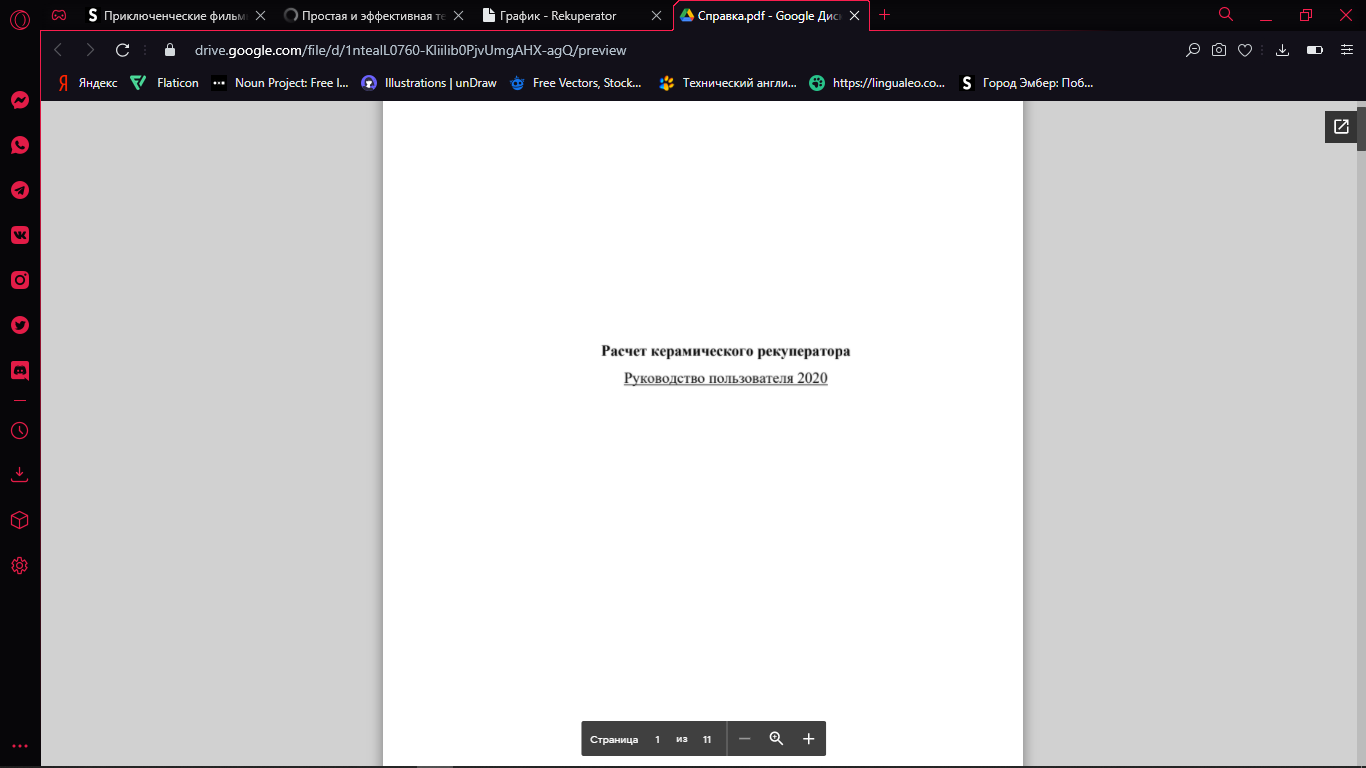


Рисунок 2.7.1 – Фрагмент методического пособия

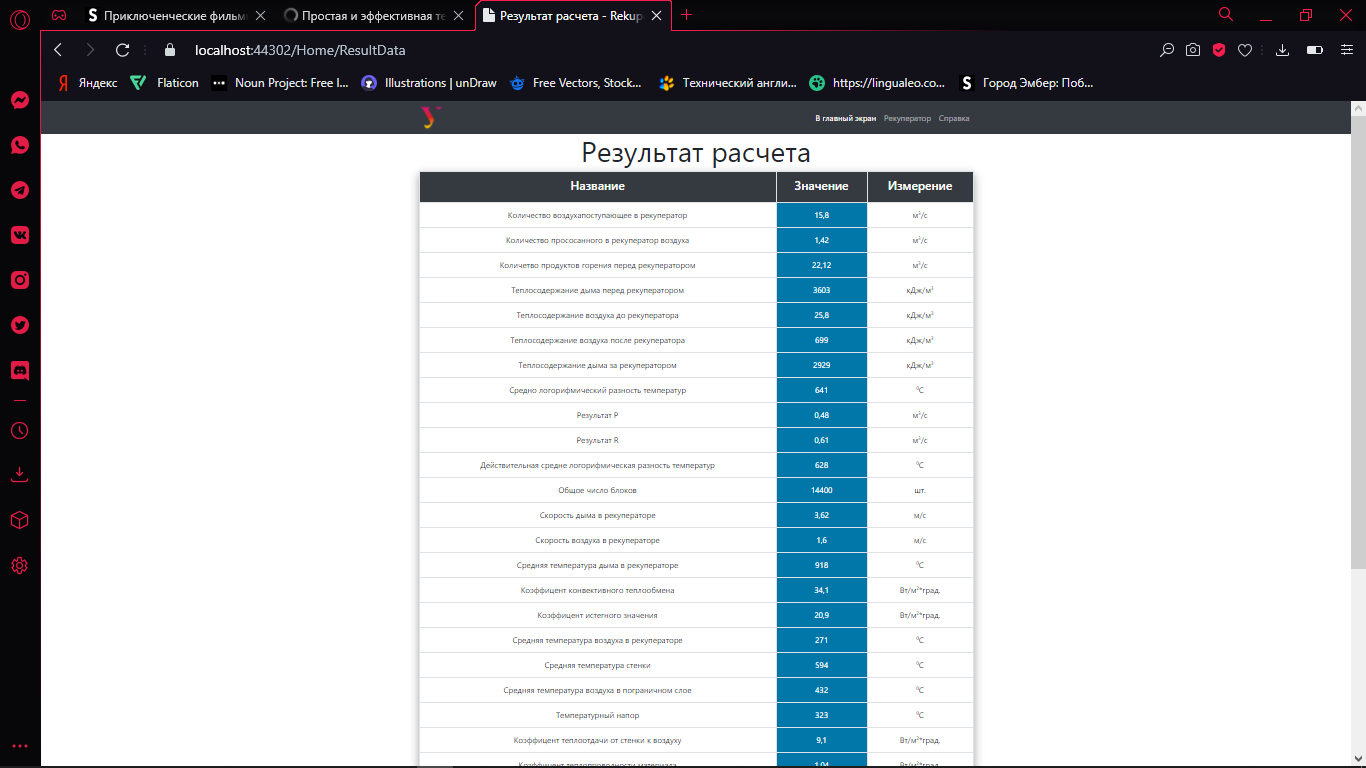
## 2.8 Создание дистрибутива

# 3 ОПИСАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

## 3.1 Установка и настройка программного средства

Данная характеристика описывает свойство ПО в части полноты удовлетворения требований пользователя и в этом смысле является определяющей для потребительских свойств ПО, в то время как остальные характеристики носят более технический характер, что не уменьшает их значение при оценке качества ПО. Кроме того, эти характеристики (такие как надежность, эффективность и др.) могут входить в число требований пользователя.

Конечный результат расчета программы показан на рисунке 3.1.1



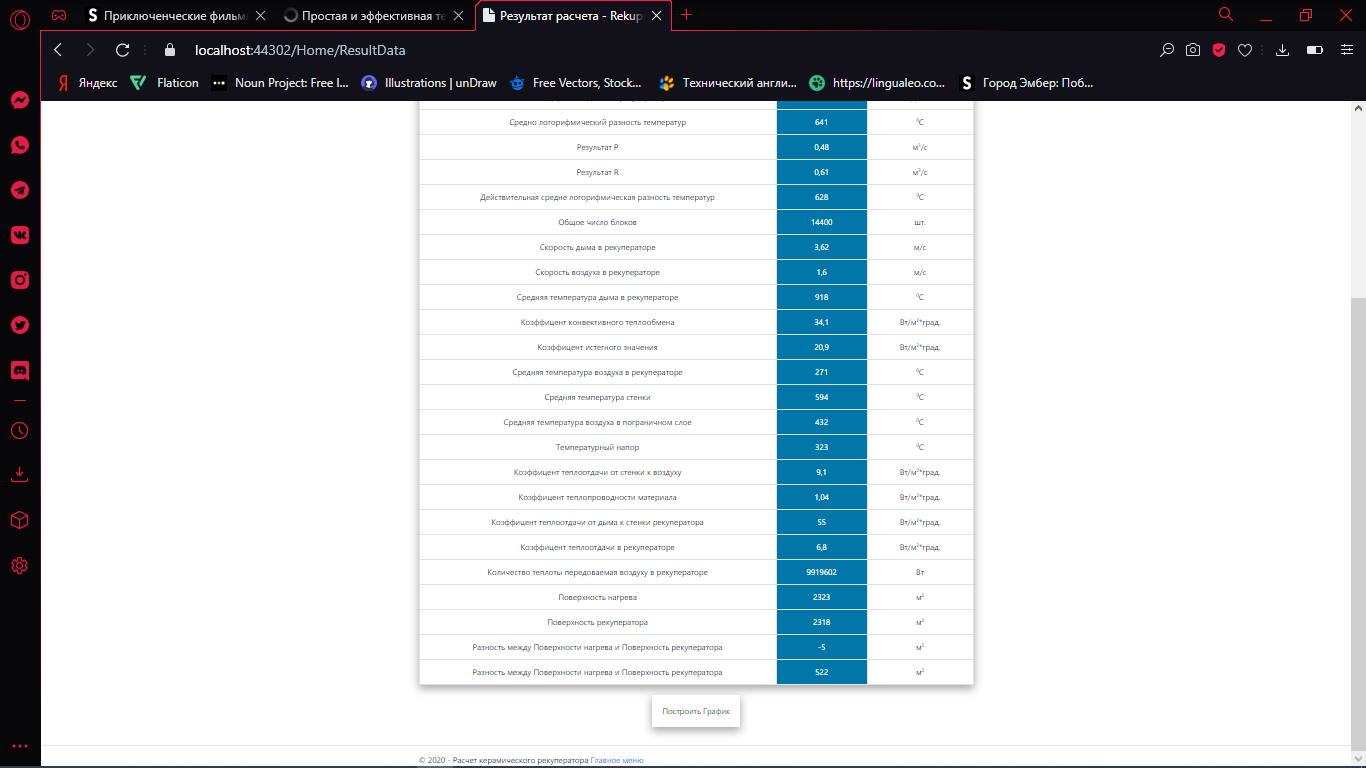
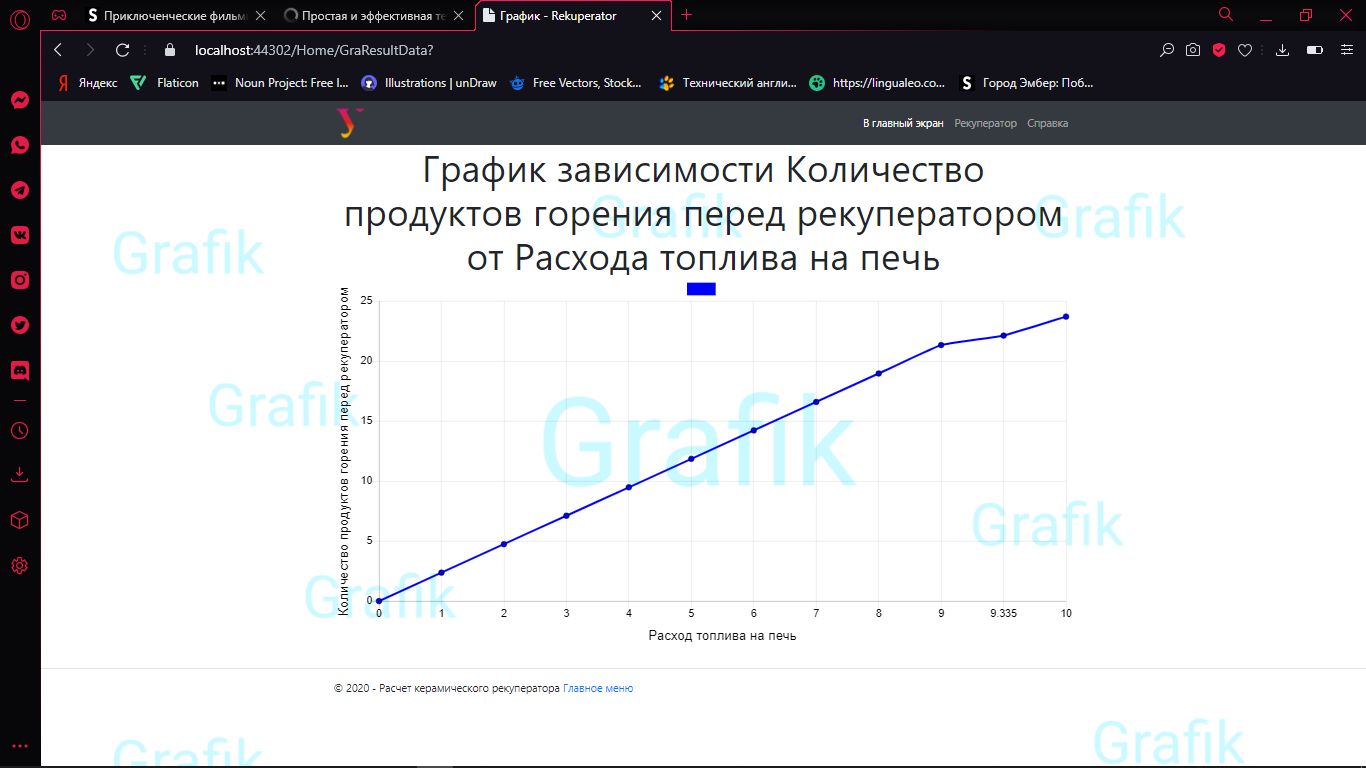


Рисунок 3.1.1 – Конечный результат расчета программы

Программное обеспечение включает функцию создания графика, пример графика показан на рисунке 4.1.2



## 3.2 Функциональные возможности программного продукта

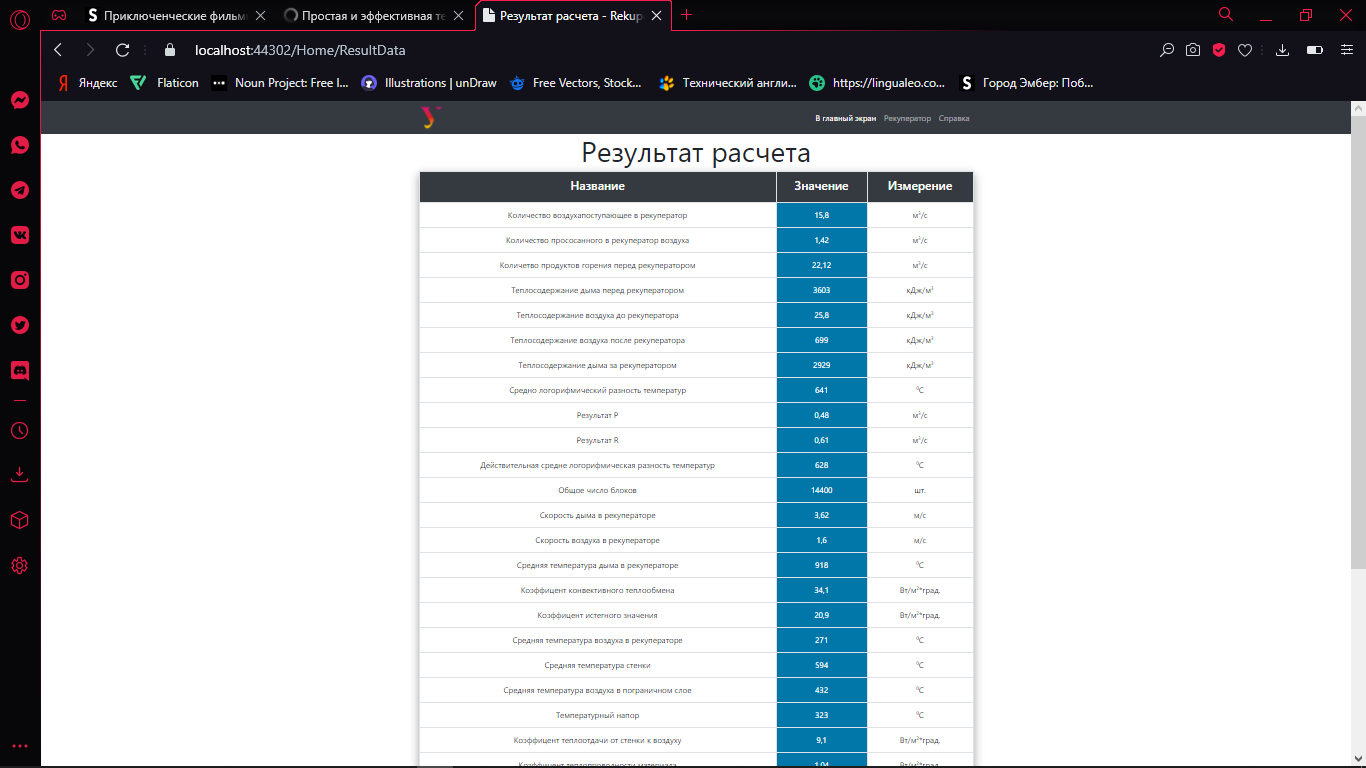
Функциональные возможности разработанной системы:

* Ввод, корректировка на форме и сохранение исходных данных для рас чета;
* Выполнение расчетов и отображение результатов в численном виде на пользовательской форме;
* Отображение результатов расчета в виде графика.

## 3.2 Технология выполнения расчетов

Функциональность: выполнение расчетов и отображение результатов в численном виде на пользовательской форме. Расчет выполняется путем обращения к математической библиотеке, файлу dll.

Пример выполнения расчета показан на рисунке 3.2.1



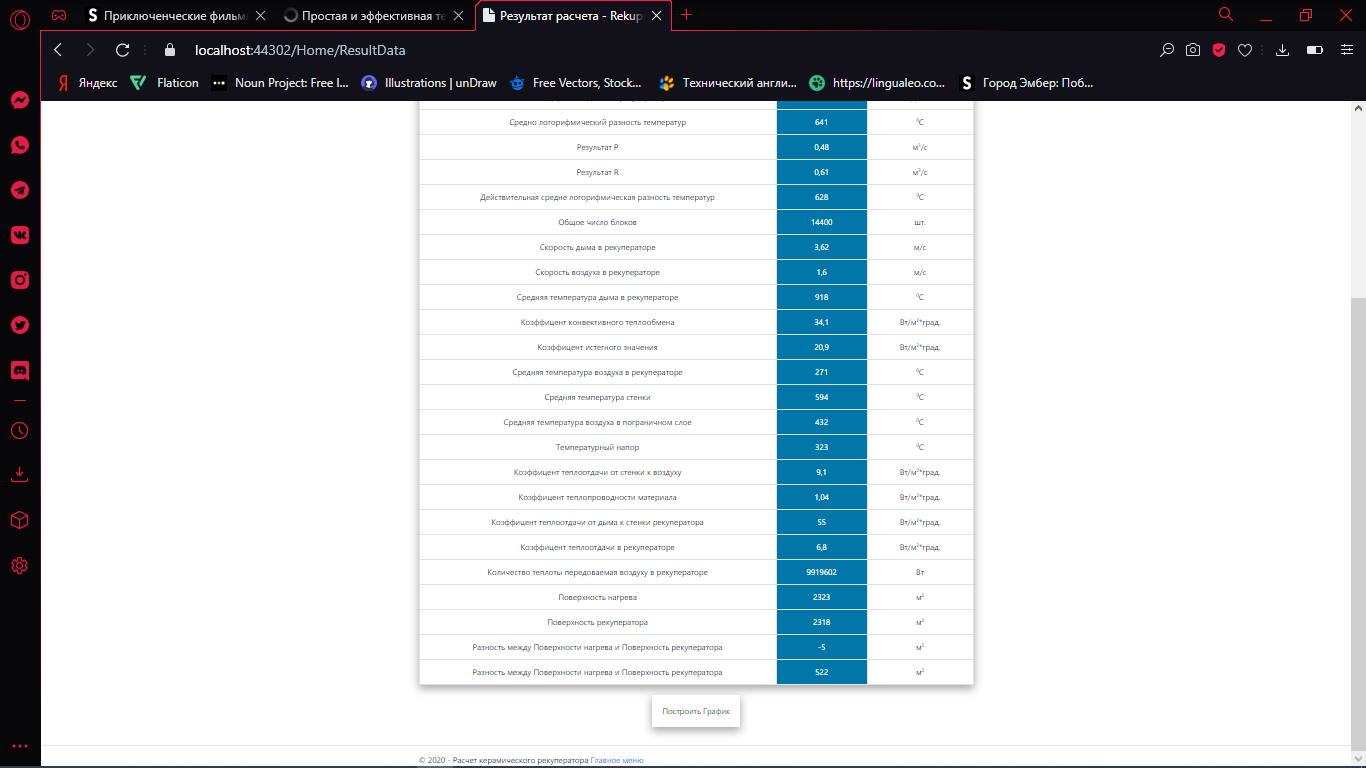


Рисунок 3.2.1– Пример выполнения расчетов программы

# ЗАКЛЮЧЕНИЕ

В ходе выполнения работы проекта по модулю, была спроектирована и разработана автоматизированная система для работы с исходными данными и выполнения расчета используя их. Данную программу можно дорабатывать и дополнять новым функционалом.

Данная версия работает, и может выходить на рынок как полностью рабочая программа, существенно облегчающей расчеты и вычисления.

# БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Павловская Т.А. C#. Программирование на языке высокого уровня: Учебник для вузов. -СПб.: БХВ-Петербург, 2007.
2. Коцюба И.Ю., Чунаев А.В., Шиков А.Н. Основы проектирования информационных систем. Учебное пособие. – СПб: Университет ИТМО, 2015. – 206 с.
3. Техника пылеулавливания и очистки промышленных газов - Г.М.-А. Алиев
4. Козлова С.А. и др. Оборудование для очистки газов промышленных печей.
5. Флёнов М.Е. Библия С#. 2-е изд. – СПб.: БХВ-Петербург, 2011. – 560 с.
6. Троелсен Э. Язык программирования C# 5.0 и платформа .NET 4.5. 6-е изд. – М.: ООО «И.Д. Вильямс», 2015. – 1312 с.
7. Горлушкина Н.Н. Системный анализ и моделирование информационных процессов и систем. Учебное пособие. – СПб: Университет ИТМО, 2016. – 120 с.
8. Шилдт Г. Польный справочник по C#: пер. с англ./ Г. Шилдт. -М: Издательский дом «Вильямс», 2006. – 752с.
9. Стасышин В.М. Проектирование информационных систем и баз данных: учебное пособие. – Новосибирск: НГТУ, 2012. – 100 с.
10. Лавров В.В. Методические указания к выполнению курсовой работы для студентов бакалавриата, обучающихся по направлению «Информационные системы и технологии» и студентов магистратуры, обучающихся по программе «Информационные системы в металлургии» / В.В. Лавров, К.А. Щипанов, А.А. Бурыкин – Екатеринбург: УрФУ, 2014. – 49 с.
11. Лошкарев Н.Б. Указания к оформлению дипломных и курсовых проектов и работ: методические указания / Н.Б. Лошкарев, А.Н. Лошкарев, Л.А. Зайнуллин. – Екатеринбург: УГТУ–УПИ, 2007. – 49 с.

# ПРИЛОЖЕНИЕ

Фрагменты листинга программного обеспечения

namespace Web\_proekt.Models

{

public class Model

{//Обозначение переменных

public double B { get; set; }

public double ALFA { get; set; }

public double LALFA { get; set; }

public double VALFA { get; set; }

public double VL { get; set; }

public double TRO { get; set; }

public double BETTAT { get; set; }

public double CV1 { get; set; }

public double TV1 { get; set; }

public double CV2 { get; set; }

public double TV2 { get; set; }

public double ETA { get; set; }

public double TD1 { get; set; }

public double TD2 { get; set; }

public double KSIDELTAT { get; set; }

public double Z1 { get; set; }

public double Z2 { get; set; }

public double Z3 { get; set; }

public double ETAD { get; set; }

public double ETAV { get; set; }

public double DDDK { get; set; }

public double ALFADL1 { get; set; }

public double KT { get; set; }

public double DDVK { get; set; }

public double S { get; set; }

public double FIBL { get; set; }

//Обозначение формул

public double Teplo\_VB1 { get; set; }

public double VBPR { get; set; }

public double VD1 { get; set; }

public double ID1 { get; set; }

public double IV1 { get; set; }

public double IV2 { get; set; }

public double ID2 { get; set; }

public double DELTAT1 { get; set; }

public double P { get; set; }

public double R { get; set; }

public double DELTAT { get; set; }

public double ZCUMMA { get; set; }

public double WOD { get; set; }

public double WOV { get; set; }

public double TD { get; set; }

public double ALFADK { get; set; }

public double ALFADL { get; set; }

public double TV { get; set; }

public double TCT { get; set; }

public double TVPG1 { get; set; }

public double DELTATNAP { get; set; }

public double ALFAV { get; set; }

public double lSH { get; set; }

public double ALFAD { get; set; }

public double K { get; set; }

public double QV { get; set; }

public double F { get; set; }

public double F1 { get; set; }

public double FRaz { get; set; }

public void Calculate()

{

CalculateBezPodbora();

Padbor();

Graph();

}

public void CalculateBezPodbora()

{

if (B == 0 || ALFA == 0 || LALFA == 0 || VALFA == 0 || VL == 0 || TRO == 0 || BETTAT == 0 || CV1 == 0 || TV1 == 0 ||

CV2 == 0 || TV2 == 0 || ETA == 0 || TD1 == 0 || TD2 == 0 || KSIDELTAT == 0 || Z1 == 0 || Z1 == 0 || Z3 == 0 || ETAD == 0 ||

ETAV == 0 || DDDK == 0 || ALFADL1 == 0 || KT == 0 || DDVK == 0 || FIBL == 0 || S == 0)

{

//Message.Show = "Это значение не может равняться нулю";

}

else

{

Teplo\_VB1 = Math.Round(1.1 \* B \* LALFA, 1);

VBPR = Math.Round(Teplo\_VB1 - (B \* LALFA), 2);

VD1 = Math.Round(VALFA \* B, 2);

ID1 = Math.Round(1716 + DELTAT \* VL);

IV1 = Math.Round(CV1 \* TV1, 1);

IV2 = Math.Round(CV2 \* TV2);

ID2 = Math.Round((VD1 \* ID1 - (1 / ETA) \* (Teplo\_VB1 - VBPR) \* (IV2 - IV1)) / (VD1 + VBPR));

DELTAT1 = Math.Round(((TD1 - TV2) - (TD2 - TV1)) / Math.Log((TD1 - TV2) / (TD2 - TV1)));

P = Math.Round((TV2 - TV1) / (TD1 - TV1), 2);

R = Math.Round((TD1 - TD2) / (TV2 - TV1), 2);

DELTAT = Math.Round(DELTAT1 \* KSIDELTAT);

ZCUMMA = Math.Round(Z1 \* Z2 \* Z3);

WOD = Math.Round((VD1 + (0.5 \* VBPR)) / ((Z1 - 1) \* 0.0274 \* (Z2 / 2)), 2);

WOV = Math.Round((Teplo\_VB1 - 0.5 \* VBPR) / (0.013 \* (Z1 \* Z3) / ETAV), 1);

TD = Math.Round(0.5 \* (TD1 + TD2));

ALFADK = Math.Round((4.617 + 0.00415 \* TD) \* ((Math.Pow(3.62, 0.8)) / (Math.Pow(0.158, 0.2))), 1);

ALFADL = Math.Round(ALFADL1 \* KT, 1);

TV = Math.Round(0.5 \* (TV1 + TV2));

TCT = Math.Round(0.5 \* (TD + TV));

TVPG1 = Math.Round(0.5 \* (TCT + TV));

DELTATNAP = Math.Round(TCT - TV);

ALFAV = Math.Round((0.9 + (0.00048 \* TVPG1)) \* ((Math.Pow(WOV, 0.2)) / (Math.Pow(DDVK, 0.5))) \* (Math.Pow(DELTATNAP, 0.1)), 1);

lSH = Math.Round(0.9 + 2.3 \* (Math.Pow(10, -4)) \* TCT, 2);

ALFAD = Math.Round(ALFADK + ALFADL, 1);

K = Math.Round(1 / ((1 / ALFAD) + (S / lSH) + (1 / ALFAV)), 1);

QV = Math.Round((Teplo\_VB1 - (0.75 \* VBPR)) \* (IV2 - IV1) \* 1000, 2);

F = Math.Round(QV / (K \* DELTAT));

F1 = Math.Round(ZCUMMA \* FIBL);

FRaz = Math.Round(F1 - F);

}

}

//График

public static double[] ArrayX;

public static double[] ArrayY;

public void Graph()

{

int razm = Convert.ToInt32(Math.Round(B + 0.5)) + 2;

ArrayX = new double[razm];

ArrayY = new double[razm];

for (int i = 0; i < razm-1; i++)

{

if (B<i)

{

ArrayX[i] = B;

ArrayY[i] = Math.Round(VALFA \* B, 2);

ArrayX[i+1] = i;

ArrayY[i+1] = Math.Round(VALFA \* i, 2);

}

else

{

ArrayX[i] = i;

ArrayY[i] = Math.Round(VALFA \* i, 2);

}

}

}

// метод подбора параметров для TV2

public void Padbor()

{

if (FRaz > 0)

{

while (TV2 < 1000)

{

TV2++;

CalculateBezPodbora();

if (FRaz <= 0)

return;

}

}

else if (FRaz < 0)

{

while (TV2 > 500)

{

TV2--;

CalculateBezPodbora();

if (FRaz >= 0)

return;

}

}

}

public ResultModel Rachet()

{

return new ResultModel

{

Teplo\_VB1 = (double)Teplo\_VB1,

VBPR = (double)VBPR,

VD1 = (double)VD1,

ID1 = (double)ID1,

IV1 = (double)IV1,

IV2 = (double)IV2,

ID2 = (double)ID2,

DELTAT1 = (double)DELTAT1,

P = (double)P,

R = (double)R,

DELTAT = (double)DELTAT,

ZCUMMA = (double)ZCUMMA,

WOD = (double)WOD,

WOV = (double)WOV,

TD = (double)TD,

ALFADK = (double)ALFADK,

ALFADL = (double)ALFADL,

TV = (double)TV,

TCT = (double)TCT,

TVPG1 = (double)TVPG1,

DELTATNAP = (double)DELTATNAP,

ALFAV = (double)ALFAV,

lSH = (double)lSH,

ALFAD = (double)ALFAD,

K = (double)K,

QV = (double)QV,

F = (double)F,

F1 = (double)F1,

FRaz = (double)FRaz,

TV2 = (double)TV2,

};

}

}