**Министерство науки и высшего образования Российской Федерации**

**федеральное государственное автономное образовательное учреждение высшего образования**

**“НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО”  
(УНИВЕРСИТЕТ ИТМО)**

**ЦЕНТР АВТОРИЗОВАННОГО ОБУЧЕНИЯ ИНФОРМАЦИОННЫМ ТЕХНОЛОГИЯМ**

**ИТОГОВАЯ АТТЕСТАЦИОННАЯ РАБОТА**

**РАЗРАБОТКА ПРОГРАММЫ ДЛЯ АКУСТИЧЕСКОГО РАСЧЕТА ПОМЕЩЕНИЯ, РАСЧЕТА РАСПОЛОЖЕНИЯ ЭЛЕМЕНТОВ АКУСТИЧЕСКОГО ОФОРМЛЕНИЯ.**

Автор\_\_\_Жуков Алексей Геннадьевич\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия Имя Отчество) (Подпись)

**Центр авторизованного обучения информационным технологиям**

Наименование программы **«Python-разработчик»**

Руководитель \_Кузьмин К.М.\_ \_\_\_\_\_\_\_\_\_\_\_\_

(Подпись)

**К защите допустить**

|  |  |  |
| --- | --- | --- |
| *Заместитель директора ЦАО ИТ,*  *к.т.н.* |  | / *Т.В. Зудилова*/ |

Санкт-Петербург, 2023г.

Обучающийся\_\_\_Жуков Алексей Геннадьевич\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Группа 124/02

(Фамилия, И. О.)

Работа принята «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2023г.

Работа выполнена с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата защиты «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2023г.

Секретарь ИАК \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия, И. О.) (подпись)

Листов хранения \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Демонстрационных материалов \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ОГЛАВЛЕНИЕ**

ВВЕДЕНИЕ 4

Цель 5

СРАВНЕНИЕ С АНАЛОГАМИ 6

Описание системы 7

ФУНКЦИИ системы 16

ТЕСТИРОВАНИЕ 18

Заключение 19

Сокращения 20

Терминология 20

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 21

**ВВЕДЕНИЕ**

В данной работе мы разработаем программу «BEDROOM STUDIO», производящую акустический расчет помещения любой прямоугольной геометрии, чтобы упростить потенциальному пользователю процесс выбора оптимальной точки прослушивания и определения расположения акустических систем (далее АС), а также панелей в данном помещении.

В наше время всё большее количество исполнителей, битмейкеров, саундпродюсеров и прочих представителей музыкальной индустрии обустраивают студию (контрольную комнату) в своей квартире, зачастую это так называемая «спальня» - помещение со сложной геометрией, дополнительными нишами, перегородками, внутренними стенами и т.д.

Идеальный вариант для выбора оптимального расположения элементов студии в таких ситуациях – измерение специальным микрофоном, который является довольно дорогостоящим профессиональным инструментом, и позволить его себе может не каждый начинающий музыкант. В связи с этим необходимо решение, которое позволит по каким-либо входным параметрам получить приблизительный оптимальный результат, или несколько результатов, показывающих как расположить элементы студии с минимальными потерями качества звука.

Программа будет строить автоматически схему помещения с координатами всех элементов студии (акустические панели, АС, точка прослушивания), предлагать несколько вариантов, если это возможно. Входными параметрами будут служить координаты углов комнаты в проекции на плоскость.

Расчетная часть программы будет делить комнату на условные зоны, имеющие параллельные стены, что является прямым фактором возникновения стоячих волн, искажающих акустическую картину. По рассчитанным стоячим волнам будут определены толщины акустических панелей, наличие в них мембраны-резонатора, что в конечном итоге будет выводиться в поля пользовательского интерфейса.

Конечному пользователю в итоге не потребуется знаний акустики помещений, правил расстановки панелей в профессиональных студиях, и прочей теории, достаточно будет ввести координаты комнаты в программу и получить данные в соответствии со своей ситуацией.

**ЦЕЛЬ:**

1. Изучить программы для расчета акустики помещений от других разработчиков, существующие на рынке.
2. Разработать программу, не требующую от пользователя специальных знаний акустики, с понятным простым интерфейсом.

**СРАВНЕНИЕ С АНАЛОГАМИ**

**1. Взгляд на предлагаемый продукт.**

Индустрия богата программами для акустического расчета помещений, но большая часть из них дорогостоящие, требуют определенного образования, имеют приличное количество функций, в которых нет необходимости специалисту в домашней студии. Мы предлагаем простое дешевое решение, выполняющее конкретную задачу – показывающее пользователю как ему расположить элементы акустического оформления в студии, при этом не вдаваясь в подробности того, как происходит расчет.

Преимущества BEDROOM STUDIO:

- низкая цена;

- занимает маленький объем памяти (менее 50 Мб);

- примитивна в использовании;

- выполняет визуализацию.

Недостатки по сравнению с профессиональными программами:

- очень узкий профиль (всего один вид расчета);

- не учитывает особенности комнаты (мебель, окна, двери и т.д.);

- низкая точность (по сравнению с измерением специальным микрофоном).

**2. Аналоги на рынке.**

2.1 CARACAD

Преимущества. Фаворит среди профессионалов. Является мощным инструментом автоматизированного проектирования, позволяет шаг за шагом конструировать акустические параметры моделируемой комнаты. Представляет из себя профессиональный софт, позволяет проводить расчеты как в 2D так и в 3D. Имеет встроенную библиотеку акустических систем, настройки модели. Моделирует анимированную переходную характеристику помещения.

Недостатки. Является сложной программой для профессионалов, требует специальных знаний, освоение требует затрат времени.

2.2 Room EQ Wizard

Преимущества. По сравнению с предыдущей является бесплатной. Работает как с геометрическими измерениями, так и с помощью измерительного микрофона. Позволяет автоматически настраивать параметрический эквалайзер для коррекции комнатных мод.

Недостатки. Отсутствует автоматический расчет акустического оформления, панели придется рассчитывать вручную, нет визуализации комнаты.

**ОПИСАНИЕ СИСТЕМЫ**

**1. Обоснование выбора необходимых технических и программных средств.**

Код программы, помимо стандартных библиотек Python, использует следующие: sqlite3, PyQt5, datetime, svg\_turtle, pyinstaller.

Библиотека sqlite3 позволяет работать с БД посредством запросов SQL. Для наших целей БД в одном файле наиболее оправдана, поскольку записей в базе будет содержаться относительно немного, это будут лишь сохраненные данные предыдущих расчетов одного пользователя. Также это позволит переносить БД с одной машины на другую, посредством копирования файла БД в каталог с программой.

Библиотека PyQt5 выбрана как оптимальная, для создания простого интерфейса. Быструю работу с данной библиотекой обеспечивает программа QtDesigner, которую мы и использовали в целях разработки дизайна. Программа позволяет внести виджеты в окно программы с помощью визуализации и конвертировать в код, что позволяет избежать временных затрат на написание параметров (габаритов, размещения, названий и т.д.) этих виджетов используя интерпретатор python.

Библиотека datetime выбрана к использованию в целях формирования уникальных значений id и имен записей в БД. Значения времени и даты шифруются при сохранении данных, что позволяет избегать дополнительных действий при создании записи.

Библиотека svg\_turtle выбрана по принципу самой лаконичной в целях создания изображения в декартовой системе координат по их значениям. Также конечное изображение выводится в векторном формате .svg, что позволяет менять его размер без потери четкости.

Библиотека pyinstaller выбрана как наиболее автономная, с учетом необходимости встраивания БД, и компиляции большей части программы в один исполняемый файл.

**2. Состав и структура системы**

Базовая версия программы состоит из следующих частей:

- модуль классов;

- модуль расчета;

- графический модуль;

- модуль интерфейса;

- модуль с текстом инструкции;

- модуль БД;

* 1. Модуль классов

Модуль классов состоит всего из двух универсальных классов, с помощью которых производятся все расчеты и проверки. Первым является класс Area(object), вторым – класс Workarea(Area).

2.1.1. Класс Area.

Класс Area является наследником класса Object и имеет следующие атрибуты:

- координата «x» начальная;

- координата «x» конечная;

- координата «y» начальная;

- координата «y» конечная;

Класс обладает несколькими методами для расчетов и проверок внутри функции engine:

- \_\_init\_\_ – инициализирует атрибуты;

- set\_coordinate – устанавливает значение заданной координаты;

- get\_coordinate – возвращает значение заданной координаты;

- exclude – устанавливает и возвращает координаты четырех точек вблизи границ зоны по четырем направлениям, для проверки по ним зоны на ее расположение относительно комнаты;

- focus – устанавливает и возвращает координаты центра зоны;

- size – возвращает значение площади зоны в метрах квадратных;

- point\_belongs – определяет, принадлежит ли заданная точка зоне, возвращает булевое значение.

- get\_lines – устанавливает и возвращает координаты линий для проверки зон;

- get\_dots – устанавливает и возвращает координаты точек для переопределения зон с погрешностью;

- prnt – выводит в консоль атрибуты зоны (для тестирования и отладки).

2.1.2. Класс Workarea.

Класс Workarea является наследником класса Area. В отличии от родительского класса имеет также атрибут direction – направление. Соответственно в нем также переопределены методы \_\_init\_\_ и prnt, и определен метод get\_direction, который возвращает направление.

* 1. Модуль расчета.

Модуль расчета представляет собой основную функцию engine, в состав которой входит алгоритм вычисления координат, для расположения элементов студии.

Рассмотрим алгоритм подробно. Входящие переменные функции engine – список координат точек углов комнаты, а также l – значение базы.

2.2.1 Формирование комнаты по координатам.

В начале алгоритма создаются два списка условных «стен», линий между заданными точками, вертикальных и горизонтальных. Каждая стена имеет три координаты – начало и конец стены по одной оси, и расположение по другой. Далее ряд функций сравнивает все стены между собой, определяет какие из них являются параллельными и образуют друг с другом зону прямых отражений. Создаются два списка зон (класс Area) – по вертикальным стенам и горизонтальным.

Очевидно, что между стенами могут возникать препятствия, а также ряд зон может образоваться за пределами комнаты, так как сравниваются все стены, каждая с каждой. В связи с этим необходимо провести ряд проверок, исключающих данные погрешности. Существует 3 типа погрешности при данном сценарии:

- зона, полностью пересекающаяся стеной или другой зоной;

- зона, частично пересекающаяся стеной или другой зоной;

- зона, определившаяся за пределами комнаты между наружными стенами.

Зона, определившаяся за пределами комнаты определяется следующим образом. Методом класса зоны устанавливается четыре точки вокруг зоны, в миллиметре за ее границами, по четырем сторонам света. Другим методом этого же класса, определяется принадлежность каждой из этих точек каким-либо другим определенным зонам. Существует лишь одна ситуация при которой зона является определившейся снаружи – если только одна из проверочных точек не принадлежит другим зонам. При этом необходимо проверить, действительно ли зона является наружной, а не полезной, для этого достаточно проверить наличие стены со стороны неопределившейся точки (одной из четырех). Делается это следующим образом, вокруг неопределившейся точки создается проверочный экземпляр класса зоны. Далее создается список проверочных точек (множество точек на каждой стене по спискам стен). Если в проверочной зоне определяется одна из этих точек, то зона полезная, иначе – зона за пределами комнаты, как следствие она исключается из списка зон. Поскольку подобные зоны могут быть соседними, проверка выполняется циклично с количеством повторений, равным четверти количества всех полученных зон.

Зона, полностью пересекающаяся стеной или другой зоной, также определяется с помощью специального метода класса зоны. Метод создает две проверочные линии от границы до границы в проверяемой зоне (в вертикальной зоне – горизонтальные сверху и снизу, в горизонтальной – вертикальные справа и слева), и проверяет, пересекают ли они какие-либо стены. Если обе линии пересекают хотя бы одну стену каждая, то зона определена ошибочно, и она также исключается из списка зон.

В случае если зона, частично пересекается стеной или другой зоной, то ее необходимо переопределить с новыми координатами. Выясняется это в том случае, если в предыдущей проверке одна из линий пересекает стену/стены, а вторая нет. Тогда эта зона отправляется в список для последующего переопределения по одному из четырех направлений. Направление определяется двумя факторами: собственным направлением зоны и позицией проверочной линии, которая не была пересечена. Рассмотрим на примере зоны, в которую стена «заходит сверху». В данной ситуации в переопределяемой зоне мы определяем какие из начальных точек углов в ней находятся с помощью все того же метода принадлежности точки зоне. Из этих точек выбираем самую нижнюю, так как зону необходимо переопределить сверху – это и будет новая координата «y» по верху зоны.

Когда все эти проверки будут выполнены и сформируются окончательные списки зон, будет определена (с помощью метода класса зоны) площадь каждой зоны. Если сумма площадей вертикальных зон равна сумме площадей горизонтальных зон, то проверки выполнены верно, и расчет продолжится далее.

2.2.2 Выбор расположения АС.

В больших помещениях рекомендуется устанавливать АС в акустических фокусах комнаты, но в условиях небольшой спальни такой возможности не представляется. Компромиссный вариант – установить АС на расстоянии четверти базы от стены. Очевидно, что е все стены комнаты могут быть достаточно длинными, для того чтобы АС поместились вдоль них, соответственно необходимо определить список возможных стен к потенциальной установке. Для начала необходимо определить значение базы. Оно может быть задано через интерфейс, если пользователь имеет предпочтения, но если оно меньше одного метра, либо больше двух, то значение определится автоматически. Чем больше площадь помещения, тем больше возможна база, например, при площади более 20 м2, база будет равна 2 м. Минимально необходимая длина стены будет равна 1,5 значения базы, поскольку до боковых стен также должно сохраняться расстояние минимум в четверть базы. В конечном итоге мы формируем список стен, удовлетворяющих данные условия.

2.2.3 Выбор рабочей зоны.

Чтобы приблизиться еще на шаг к конечному списку вариантов, необходимо отфильтровать список стен, исключить те, у которых рабочую зону (далее РЗ) организовать невозможно из-за препятствий.

Первым делом мы создаем потенциально возможные РЗ (класс Workarea) по две с противоположных сторон каждой стены, формируем их список и сразу исключаем те, геометрический центр которых находится вне комнаты (не определяется ни в одной из зон). В таком случае мы далее работаем со списком зон у каждой потенциальной стены. РЗ имеет определенные габариты, связанные со значением базы, и разумеется внутри нее е должно быть препятствий. Поскольку у нас уже есть список проверочных точек вдоль каждой стены, используем его, чтобы проверить каждую РЗ. Исключаем из списка РЗ зоны в которых есть препятствия и переходим к следующей проверке.

2.2.4 Размещение фронтальных и боковых акустических панелей.

Как правило для оптимальной акустической картины в жилом помещении достаточно двух фронтальных акустических панелей (далее АП), и четырех боковых. Фронтальные АП устанавливаются непосредственно за АС для устранения ранних отражений, боковые устанавливаются по оси точки прослушивания, а также по оси в три четверти расстояния от точки прослушивания до оси АС.

В каждой потенциальной рабочей зоне мы определяем точки установки АП. Для этого в зонах определяются расстояния до необходимых осей, а в центрах осей определяется зона, которой они принадлежат. Так мы можем определить координаты стен, для расположения АП. После определения центров АП создаем вокруг них проверочные зоны, и проверяем не содержат ли они начальных точек углов комнаты, поскольку панель устанавливать на углы не всегда уместно. Если все панели на своих позиция установить возможно, то такая зона добавляется в список предварительных вариантов, остальные исключаются.

2.2.5 Размещение тыловых панелей.

Алгоритм начинается с определения количества АП. Для разной ширины зоны будет разное количество АП, от двух до четырех. В случае если за РЗ нет ровной стены и установить все необходимые АП невозможно, допускается установить все возможные АП, но не менее двух, при любой ширине РЗ.

После утверждения начального количества АП, определяются оси их центров, а также границ. По положениям осей определяем (по аналогии с боковыми панелями) зоны, в которых они находятся, а соответственно и границы для их установки. Таким же образом создаем проверочные зоны по их центрам, проверяем, принадлежат ли им начальные точки углов. Если по меньшей мере две панели установить возможно без препятствий, то в список окончательных вариантов добавляется словарь с данными координат зоны, точки прослушивания, АС и всех АП.

2.2.6 Расчет параметров АП.

Последним шагом расчета является определение параметров АП: толщины и наличии мембраны. Есть несколько значений стоячих волн при которых используется поглотитель определенной толщины. Чем ниже частота волны, тем толще поглотитель. Панели в данном проекте используются одинаковые: размером 0,6 на 1,2 м с мембраной из фанеры плотностью 10 кг/м2. Определить необходимую толщину панели можно по графику зависимости толщины поглотителя от частоты, полученному по нескольким известным значениям путем аппроксимации функции одной переменной [11]. Промышленный поглотитель изготавливается и продается в пластинах толщиной кратной 50 мм, поэтому АП будут иметь конечную толщину с запасом в большую сторону.

Рассчитываем стоячие волны по осям всех АП, кроме фронтальных (они фиксировано должны быть 100 мм без мембраны), и определяем толщину каждой. Если частота достаточно высокая, то мембрана не требуется. Значение толщины добавляется в словарь соответствующей АП в списке окончательных вариантов, если мембраны нет, то толщина записывается как «0».

* 1. Графический модуль.

Графический модуль состоит из трех базовых функций. В первой функции описан алгоритм компиляции изображения схемы, во второй – алгоритм создания файла изображения. Третья функция – main, исполняющая первые две.

Построение изображения действует в следующем порядке. По координатам точек с заданным смещением курсор огибает контур комнаты, в следствии чего мы получаем проекцию ее геометрии. Данные о размещении всех элементов передаются в функцию модулем интерфейса. Далее, курсор несколько раз поднимается, перемещается и опускается, для того чтобы отобразить точки прослушивания, АС и АП. В процессе отображения АП на схеме в каждую из них печатается число – ее порядковый номер.

Каждая новая схема заменяет собой предыдущую в одном и том же файле, для экономии места на жестком диске, а также для упрощения алгоритма.

* 1. Модуль интерфейса.

Модуль интерфейса (рисунок 1) состоит из двух классов: первый – моделирует окно интерфейса и органы управления, второй только отображает дизайн фона.

Первый класс, помимо методов, организующих вид полей и кнопок, также имеет ряд функциональных. Рассмотрим их более подробно.

Метод count: формирует список точек, а также значение базы (если оно введено), возвращая значения заполненных полей, вызывает модуль расчета, передает ему переменные.

Метод alternative: выбирает первый и каждый следующий вариант из списка возможных, вызывает графический модуль, передает ему переменные, выводит в интерфейс результаты расчета, открывает схему в отдельном окне.

Метод formpanel: формирует текст вывода результатов расчета каждой панели.

Метод reset: сбрасывает текущий результат, очищает поля ввода/вывода, закрывает окно со схемой, если оно было открыто.

Метод setup: обращается к записи в БД, формирует значения координат, вводит их в соответствующие поля.

Метод instruction: сбрасывает результат, очищает все поля, выводит инструкцию.

Метод saving: обращается к БД, шифрует текущие значения, записывает их в БД.

Метод showsaves: обращается к БД, выводит записи БД в специальное поле.

|  |
| --- |
|  |
| Рисунок 1. Интерфейс |

* 1. Модуль с текстом инструкции

Простой модуль, содержит всего одну переменную с типом данных – строка. В строке прописан текст инструкции для пользователя, который по умолчанию при запуске программы выводится в интерфейс. Инструкцию в интерфейсе возможно открыть и закрыть по желанию пользователя.

2.6 Модуль БД

База данных в нашей программе довольно лаконична, представляет собой файл SQLITE3 с единственной таблицей SAVEDDOTS.

Таблица SAVEDDOTS состоит из трех столбцов (в скобках указан тип данных SQL): id (BIGINT), name (TEXT), data (TEXT).

Все SQL запросы выполняются кодом программы автоматически.

**3. Руководство пользователя.**

Действия пользователя программы описаны в инструкции, внедренной в интерфейс. Текст инструкции представлен далее:

*Добро пожаловать в BEDROOM STUDIO!*

*Данная утилита выполняет расчет положений акустических панелей, точки прослушивания и акустических систем (далее АС), основываясь на геометрии Вашей комнаты. Чтобы произвести расчет, следуйте данной инструкции.*

*Элементы интерфейса:*

*- поля ввода координат: предназначены для ввода двухмерных координат улов комнаты целые или дробные числа, примеры "1, 2.0, 3.3...");*

*- поле ввода l: предназначено для ввода базы - расстояния между АС (опционально), если оставить поле пустым, значение базы будет выбрано исходя из площади комнаты;*

*- поле информации: в этом поле Вы читаете данную инструкцию, также здесь будут отображены результаты после расчета;*

*- кнопка "Инструкция": очищает поля ввода/вывода, вызывает данную инструкцию;*

*- кнопка "Расчет": выполняет расчет по заданным координатам;*

*- кнопка "Сброс": очищает поля ввода/вывода;*

*- кнопка "Альтернативный вариант": меняет вариант расположения, если есть другие возможные.*

*- поле "Предыдущие варианты": содержит записи с датой и временем предыдущих расчетов для повторного автоматического ввода координат.*

*После успешного расчета открывается окно со графической схемой расположения элементов в комнате.*

*Обозначения схемы:*

*- зеленый круг: точка прослушивания;*

*- синий круг: АС, первая и вторая;*

*- черный квадрат с числом: акустическая панель и ее порядковый номер;*

*Порядок действий:*

*1. Измерьте Вашу комнату. Представив комнату сверху, угол, левее или ниже которого комната не продолжается, это будет координата 'x'=0.0-'y'=0.0. Она уже указана в программе первой. Ось X располагается вертикально, Y - горизонтально. Определите координаты остальных углов комнаты, двигаясь по часовой стрелке. Значения координат следует вводить в этой же последовательности, единицы - метры.*

*ВНИМАНИЕ! Программа работает только с комнатами, имеющими исключительно прямые углы!*

*2. Для того чтобы начать новый расчет, введите значения в поля ввода координат. Если необходимо вновь произвести ранее выполненный расчет, выберите его в поле "Предыдущие варианты" и откройте двойным кликом левой кнопки мыши по соответствующей записи, значения координат будут выставлены автоматически;*

*3. Если необходимо использовать фиксированное значение базы, введите его в поле ввода l, либо оставьте его пустым;*

*4. Нажмите кнопку "Расчет". В полях вывода появится информация о размещении элементов и параметрах панелей, также откроется окно со схемой, а в поле "Предыдущие варианты" будет автоматически помещена новая запись текущего расчета. Если расчет не произошел, проверьте введенные данные;*

*5. Нажмите кнопку "Альтернативный вариант", чтобы просмотреть прочие варианты в рамках расчета.*

*Приятного использования!*

**4. Практическая значимость работы.**

Программа разрабатывалась в качестве инструмента экономии времени для людей, работающих с музыкой в том или ином ключе. Знания акустики помещений для расчета студии вручную получить самостоятельно и быстро довольно трудно, поэтому мы предлагаем данное решение, чтобы технический аспект не обременял творчество для таких людей. Оно окажет им помощь в получении быстрого результата без лишних размышлений и временных трат, позволив сосредоточиться на своей работе.

**ФУНКЦИИ СИСТЕМЫ**

В данном разделе перечислены функции системы с детализацией:

1. Нажатие на кнопку «Расчет»:
   1. Формирование данных из полей ввода, получение результата (метод count);
      1. Расчет вариантов расположения элементов студии (функция engine);
         1. Создание стен (функция walls);
         2. Создание зон (функция wall\_selection);
            1. Формирование зон из параллельных стен (функция wall\_overlap\_comparison);
         3. Создание проверочных точек вдоль стен (функция add\_check\_points);
         4. Проверка зон на наличие препятствий (функция empty\_check);
         5. Формирования списка зон за пределами комнаты (функция ex\_areas);
         6. Формирование списков стен для расположения АС (функция wall\_length);
         7. Создание потенциальных РЗ (функция wa\_filter);
         8. Формирование списков зон в пределах комнаты (функция area\_pass);
         9. Определение центров расположенных панелей (функция pan\_center);
            1. Определение зон панелей (функция area\_search);
            2. Определение координат панелей (функция c\_3456);
         10. Определение количества тыловых панелей (функция side\_panels);
             1. Определение координат панелей по вертикали (функция verticals);
             2. Определение координат панелей по горизонтали (функция horizontals);
         11. Формирование результатов (функция get\_results);
             1. Определение координат АС (функция add\_AS);
             2. Определение параметров панелей (функция membrane);
   2. Формирование текста вывода результатов, вывод схемы (метод alternative);
      1. Формирование изображения схемы (функция graphic);
         1. Начертание схемы (функция draw\_room);
            1. Начертание элемента (функция draw\_element);
         2. Сохранение файла рисунка (функция write\_file);
         3. Запуск функций создания изображения (функция main);
   3. Сохранение результата в БД (метод saving);
2. Нажатие на кнопку «Альтернативный вариант»:
   1. Описано в пункте 1.2 (метод alternative);
3. Нажатие на кнопку «Сброс»:
   1. Очистка полей ввода/вывода, закрытие окна схемы (метод reset);
4. Нажатие на кнопку «Инструкция»:
   1. Открытие инструкции (метод instruction);
      1. Очистка полей ввода/вывода, закрытие окна схемы (метод reset);
5. Двойное нажатие левой кнопкой мыши на запись в поле « Предыдущие варианты»:
   1. Очистка полей ввода/вывода, закрытие окна схемы (метод reset);
   2. Автоматическое заполнение полей ввода данными из БД (метод setup);

**ТЕСТИРОВАНИЕ**

Алгоритм построения помещения, в процессе разработки, получил несколько этапов фильтрации различных погрешностей. Все возможные исключения стали основой для построения функций, уточняющих данный алгоритм, что является следствием работы одного и того же простого автоматического теста.

Порой жилые комнаты имеют весьма непредсказуемые конфигурации, поэтому, на этапе тестирования, было необходимо получить возможность автоматического случайного заполнения вводных данных координат и последующего выполнения алгоритма.

Рассмотрим структуру теста подробнее.

Количество точек (а соответственно и количество стен) выбиралось случайно, с использованием модуля random.randint, с условием: четное, не более 30, но не менее 4, так как помещение должно быть по меньшей мере квадратное с четырьмя стенами.

Далее формировался список из этих случайных чисел. Первая и вторая позиции заменялись значениями «0», для шага из начала координат. Затем следовал цикл заменяющий значение в двух чисел через два следующим: 6 и 5 позиции списка заменялись на равные 4 и 3 позициям соответственно, 7 и 8 – оставались без изменений, далее 10 и 9 заменялись на равные 8 и 7, и так далее, пока список не закончится. Последняя позиция заменялась на значение «0», для того чтобы «замкнуть» комнату. Завершающим шагом была проверка на пересечение стен, функцией сравнения координат, примененной в конечном итоге в модуле расчета: если стены не пересекались, то значит комната реалистичная. В этом случае список координат, сформированных тестом, отправлялся переменой в функцию, и она выполнялась.

Бесконечный цикл такого быстрого теста позволял выполнять алгоритм программы до тех пор, пока не появится ошибка в построении зон. В то же время в алгоритме программы после каждого подмодуля на печать выводится маркер выполнения, что при возникновении ошибки позволяло быстро отследить, в каком именно месте необходимо внести изменения. Разработка каждого исключения определялась анализом координат, после ввода которых алгоритм ломался.

Недостаток данного теста заключается в том, что он не учитывает отрицательные координаты, работает только в первом квадранте декартовой системы координат, но мы можем доработать это в перспективе.

Другие модули программы имеют более простые алгоритмы, поэтому, как и пользовательский интерфейс, тестировались вручную на примерах с одним, двумя и четырьмя вариантами расположения элементов студии.

**ЗАКЛЮЧЕНИЕ**

Разработанное приложение свернуто в один файл (+ файл базы данных) и выполняется в системе Windows без участия интерпретатора, не требует установки Python. Занимаемая память в распакованном виде 48,8 Мб, что в целом позволяет обходиться без инсталлятора, и распространять его в виде архива.

Достигнутые цели:

1. Расчет выполняется без ошибок;
2. Интерфейс достаточно прост для пользователя с любым уровнем подготовки;
3. Каждый расчет сохраняется в БД автоматически;

В перспективе:

1. Провести тесты с отрицательными координатами, доработать алгоритм;
2. Добавить функции расположения окон, дверей и других элементов, для их учета в расчете;
3. Добавить разный материал АП, разные виды мембран;
4. Добавить угловые ловушки.

Возможные модификации:

1. Расчет по текущему алгоритму в трехмерной среде, учет высоты помещения;
2. Добавление функции измерения микрофоном, с использованием аудиоинтерфейса;
3. Учет в расчете моделей АС. Встроенная библиотека моделей;
4. Перенос в виде приложения на системы Android и iOS.

**СОКРАЩЕНИЯ**

АП – акустическая панель

АС – акустическая система

БД – база данных

РЗ – рабочая зона

**ТЕРМИНОЛОГИЯ**

База – расстояние между акустическими системами, а также расстояние между каждой из них до точки прослушивания.

Акустический фокус – точка в прямоугольном помещении, находящаяся от каждой стены на расстоянии в 33% от длины направления.

Стоячая волна – звуковая волна, возникающая между параллельными препятствиями, определенной длины, которая при отражении входит в противофазу и формирует акустические искажения.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Вахитов Ш.Я., Ковалгин Ю.А., Фадеев А.А., Щевьев Ю.П. Акустика: Учебник для вузов – М.:Горячая линия,2009. – 660 с.: ил. Стр. 12 - 1.3 Сложение звуковых волн. Стр. 257 – Глава 4 Акустика помещений.
2. Любанович Б. Простой Python. Современный стиль программирования. – СПб.:Питер,2016. – 480 с.: ил. – (Серия «Бестселлеры O’Reilly»). Стр.70 – Глава 3.
3. Свейгарт Э. Чистый код для продолжающих. - СПб.:Питер,2022. – 384 с.: ил. – (Серия «Библиотека программиста»). Стр.03 – Классы, которые должны быть функциями или модулями. Стр.104 - Списковые включения внутри списковых включений.
4. Егоров А. Модуль turtle. Создаем графику в Python: [Электронный ресурс]. URL: https://www.youtube.com/playlist?list=PLQAt0m1f9OHvowenYcOHrRP\_v1VN-0TWF
5. Плугарь И.В. Конвертируем .py в .exe | PyInstaller: [Электронный ресурс]. URL: https://www.youtube.com/@plugar\_inf/about
6. Тищенко Д. SQLITE 3 Python [ 1 ] | Создание таблицы, добавление данных, проверка и вывод: [Электронный ресурс]. URL: https://www.youtube.com/watch?v=mRuylFh9wfY&t
7. Поздеев К. Основные команды SQL, которые должен знать каждый программист: [Электронный ресурс]. URL: https://tproger.ru/translations/sql-recap
8. Мантей А. Simple way to display SVG image in a PyQt window: [Электронный ресурс]. URL: https://stackoverflow.com/questions/35129102/simple-way-to-display-svg-image-in-a-pyqt-window
9. Qt Designer Manual: [Электронный ресурс]. URL: https://doc.qt.io/qt-6/qtdesigner-manual.html
10. Модуль datetime: [Электронный ресурс]. URL: https://pythonworld.ru/moduli/modul-datetime.html
11. Калькулятор аппроксимации функции одной переменной: [Электронный ресурс]. URL: https://planetcalc.ru/5992/
12. Калькулятор панельных поглотителей: [Электронный ресурс]. URL: https://acoustic.ua/forms/calculator6.html