

Создание четырёхмерного лабиринта

Автор: **Гладышев Михаил Александрович**, 11 кл., МБОУ Лицей № 136 Ленинский района г. Новосибирска

Научный руководитель: **Валюхова Светлана Викторовна**, учитель информатики высшей квалификационной категории

Актуальность.

Во всех современных приложениях используются графические двухмерные интерфейсы для работы с пользователем, а так же трёхмерная графика.

Я поставил себе задачу освоить использование графики в программах, написанных своими руками.

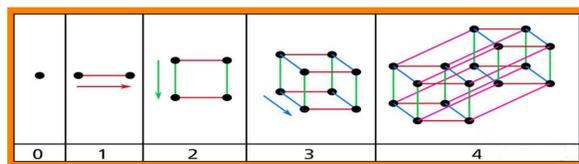
Как способ решения проблемы я выбрал создание игры лабиринта в псевдо четырёхмерном пространстве.

В современной науке часто используется многомерность для решения различных задач.

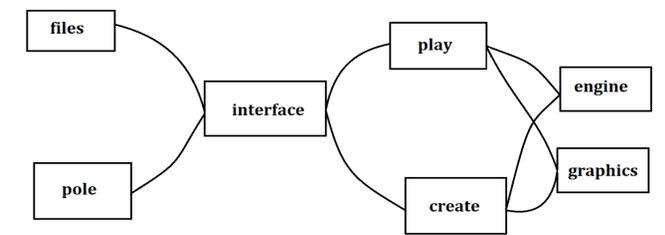
Четырёхмерное пространство

Почему это важно? Зачем понадобились такие пространства? Во-первых, четырехмерное пространство нам важно, потому что оно применяется в физике, потому что мы в нем живем. А зачем нужны пространства более высоких измерений? Давайте представим себе, что мы изучаем какие-то объекты, которые обладают большим количеством параметров. Например, мы изучаем страны, и у каждой страны есть территория, количество населения, внутренний валовой продукт, количество городов, какие-нибудь коэффициенты, индексы, что-нибудь такое. Мы можем представлять себе каждую страну в виде одной точки в каком-то пространстве достаточно высокой размерности. И оказывается, что с математической точки зрения это правильный способ об этом думать.

Представить себе четырех-, пяти- или шестимерное пространство довольно сложно, но математики не боятся трудностей, и им мало даже стомерных пространств. Математики придумали бесконечномерное пространство — пространство, содержащее бесконечное количество измерений. В качестве примера такого пространства можно привести пространство всех возможных функций, заданных на отрезке или прямой.



- У линейной алгебры есть многочисленные приложения не только в математике, но и в самых разных науках, начиная с физики и заканчивая, например, экономикой или политической наукой. В частности, линейная алгебра является основой для многомерной статистики, которая как раз используется для вычленения связей между различными параметрами в каких-то массивах данных. В частности, популярный ныне термин **Big Data** зачастую связывается с решением задач по обработке данных, которые представляются именно большим количеством точек в пространстве какой-то конечной размерности. Чаще всего такие задачи можно переформулировать и разумно воспринимать именно в геометрических терминах.
- Для более лёгкого представления четырёхмерного пространства, можно провести аналогию с большим количеством комнат одинакового размера, но с разным расположением мебели. Вы можете мгновенно перейти из одной комнаты в другую сохраняя такие же координаты как и в изначальной, но вы не сможете перейти если на вашем месте находится какая-то мебель.
- Алгебраически** четырёхмерное пространство может быть построено как множество **векторов** с четырьмя вещественными **координатами**.
- Геометрически** в простейшем случае четырёхмерное пространство рассматривается как **евклидово пространство** четырёх измерений, в более общем рассмотрении оно имеет **неевклидову метрику**, переменную от точки к точке.
- Для визуализации представления четырехмерного пространства была написана программа на языке программирования Python.
- Для облегчения понимания и написания кода, была выбрана алгебраическая модель понимания четырёхмерного пространства.
- Для вывода на экран четырёхмерной фигуры будет использована её трёхмерная проекция в данной координате, что является кубом.
- В проекте было использовано одно свойство многомерных пространств, что объект ниже по количеству измерений не сможет выйти за пределы объекта с большим количеством измерений, а будут всегда возвращаться на исходную позицию.



2.2 files

В данном блоке, пользуясь встроенными функциями python, программа обращается к файлам необходимым ей в данный момент или создаёт новый в случае с созданием новых уровней.

Для программы нужны файлы уровней, которые хранятся в формате txt и могут считываться программой. В файлах уровней находятся несколько важных аспектов: 1. Размер уровня 2. Разметка уровня 3. Координаты старта 4. Координаты финиша.

В отдельном файле так же находится список всех уровней, к которому программа обращается для вывода его игроку, для проверки существования уровня с таким же названием и для удаления информации об удалённом уровне.

2.3 pole

Данный блок представляет собой несколько классов: класс клетки, класс игрового поля и класс редактируемого поля.

Класс клетки представляет собой ячейку, которая хранит свои координаты и своё состояние. Данный класс нужен в большей части для создания ячеек с большим функционалом, нежели простые стены.

Класс игрового поля это массив, формирующийся из данных которые он получил при вызове. В нём хранится массив заданных размеров, который состоит из объектов класса ячейки

Класс редактируемого поля это пустой массив заданного размера, состоящий из объектов класса ячейка. Главное отличие от игрового поля состоит в возможности сохранять данные и менять уже имеющиеся.

2.4 engine

В этом блоке находится большинство функций взаимодействия программы с игроком, а конкретнее все функции управления игрой при помощи клавиатуры и все расчеты, связанные с ними.

Программа проверяет, может ли игрок переместиться в выбранном направлении или же там находится стена. Если игрок перемещается дальше максимальной координаты, то его пытаются переместить на нулевую точку по выбранной игроком координате.

Так же в игре присутствует функция перемещения камеры. При помощи клавиатуры игрок может отдельно перемещать камеру в 4 измерениях, или посмотреть на слои ниже или выше того, на котором он находится.

Считывание нажатия клавиш клавиатуры происходит при помощи библиотеки pygame.

2.5 graphics

В блоке graphics находятся все команды связанные с отрисовкой игрового поля и всех необходимых для этого элементов. Для изображения 3-х мерной графики была использована библиотека ruopengl.

Основными объектами на поле являются кубы разных цветов: серый – стена, красный – игрок, зелёный – финиш.

Для обучения пользования данной библиотекой была использована методичка представленная на сайте:

<https://pythonist.ru/vvedenie-v-opengl-i-pyopengl-chast-i-sozдание-vrashayushhegoya-kuba/?ysclid=lc1x2cwfxx241218821>

2.6 create

В данном блоке идёт рекурсивное повторение функций необходимых для функционирования окна редактора уровней:

Перевод окна в состояние, необходимое для работы opengl, считывание действий игрока, обновление картинки на экране

2.7 play

В данном блоке идёт рекурсивное повторение функций необходимых для функционирования окна игры:

Перевод окна в состояние, необходимое для работы opengl, считывание действий игрока, обновление картинки на экране

2.8 interface

В этом блоке создаются все меню взаимодействия с пользователем:

Основное меню, меню play, меню create, и меню появляющиеся при завершении уровня или нажатии клавиши Esc.

В каждом меню находятся несколько объектов класса кнопка по нажатию на которые происходит действие. В классе кнопка находится информация о картинке, которой она отрисовывается на экран, та функция, за которую она отвечает, и функция которая определяет нажатие по кнопке. При наведении курсором на кнопку она уменьшится в размере, показывая, что на неё можно нажать.

Основное меню состоит из 3 кнопок. Кнопки play и create служат для перехода в соответствующие меню, а кнопка Exit, для выхода из игры.

В меню play находятся названия уровней, которые можно выбрать. Для возможности выбора уровня была создана функция преобразующая текст с названием в кнопку и выделяющая соответствующую строчку при её нажатии. Так же в данном меню находятся кнопки play и delete, отвечающие за начало игры и удаление уровня соответственно. Последняя кнопка back отвечает за возврат в основное меню

В меню create находятся 7 кнопок. Кнопки name, x, y, z, w отвечают за ввод параметров для создания нового уровня. Для них создана функция ввода текста которая позволяет выводить на экран, набираемый с клавиатуры текст в реальном времени, а так же его удаление.

Для полей кнопок x, y, w, z созданы ограничения, из-за чего в них можно вводить только цифры и нельзя ввести число больше 13, данное число было выбрано из соображений, что уровни большого размера могут плохо отображаться на экране, и будут неудобными к прохождению. Все эти 5 кнопок будут активны до нажатия клавиши Enter, из-за чего вы не сможете совершать другие действия. Кнопка back отвечает за возврат в основное меню. Кнопка create при нажатии создаёт редактируемый, полностью пустой уровень, при условии, что все другие заполняемые поля, заполнены.

В меню, которые появляются при завершении уровня или нажатии клавиши Esc находятся кнопки Exit, возвращающие в основное меню, кнопка Continue, позволяющая продолжить создавать или играть в уровень, и кнопка Save благодаря которой можно сохранить созданный уровень и выйти в основное меню.

Все основные кнопки являются картинками, и посмотреть их внешний вид можно в приложении. Для их создания использовалась программа Adobe Photoshop CS6.

Так как я не нашёл способа размещения картинок с учётом расширения экрана, все кнопки были установлены под имеющиеся у меня расширения 1920x1080, из-за чего на другом расширении монитора программа будет работать некорректно.