3D визуализации 2D плана помещения методом бросания лучей

Выполнил: студент 05230 гр., Шорников А. Е.
Научный руководитель — к.ф.-м.н., ст. преп. **Трунин Дмитрий Олегович**
Научный консультант — вед. пр. ЛПС БГУ **Брагин Александр Фёдорович**

Бурятский государственный университет
Институт математики и информатики
Кафедра прикладной математики

Улан-Удэ
2017г.

Введение

Бурятский государственный университет

Институт математики и информатики

- **Улан-Удэ, Ранжурова, 5**
- У организации 22 филиала
- Сегодня 08:00–17:00, обед 12:00–13:00
  Откроется завтра в 08:00

Сегодня 08:00–17:00 Закрыто
обед 12:00–13:00

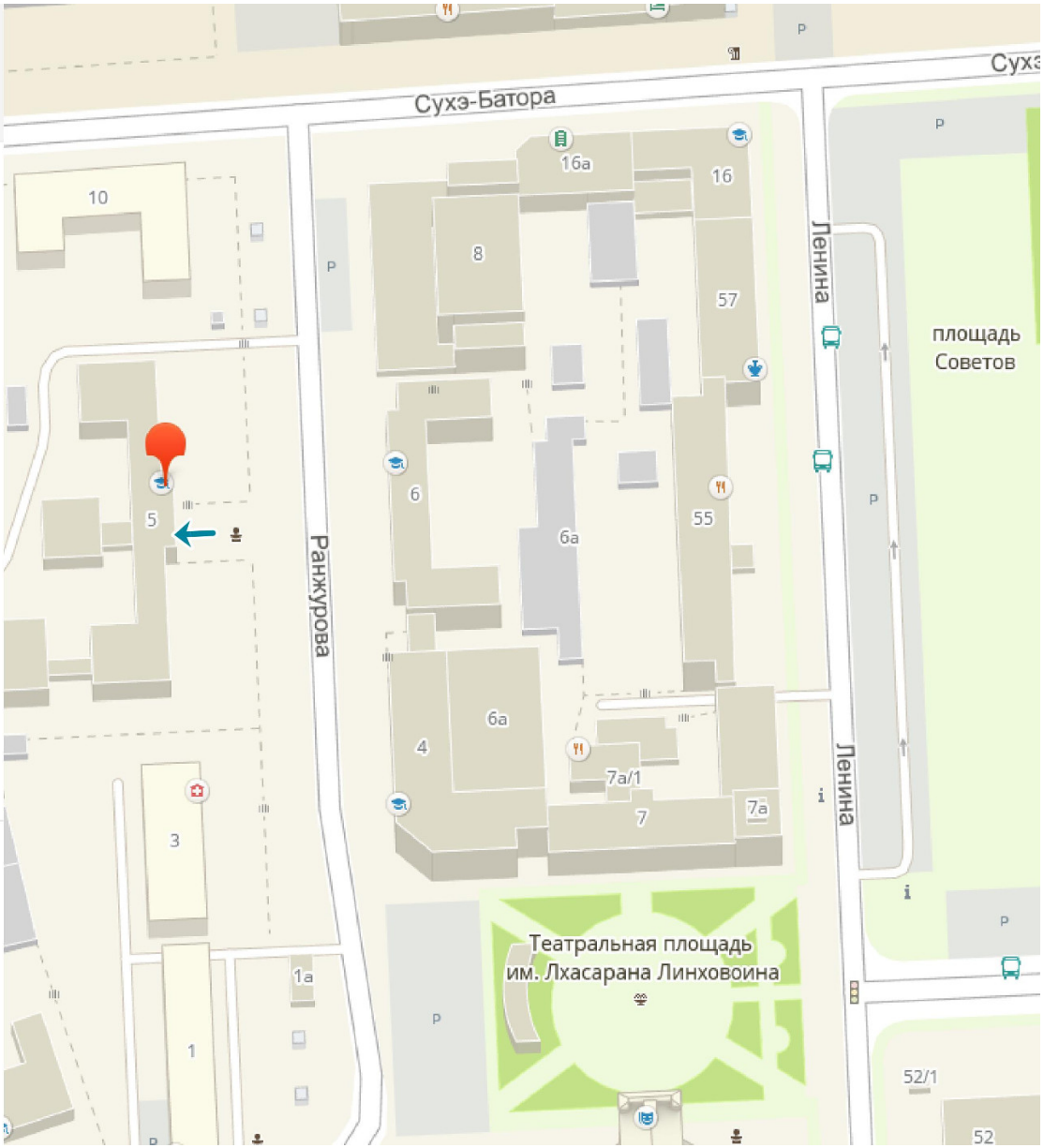
Улан-Удэ, Ранжурова, 5

1 корпус; 1203, 1204 кабинет

670000

- 3 этажа
- пл. Советов — 350 м
- Найти парковки рядом

Университеты



Цели и задачи

Цель работы

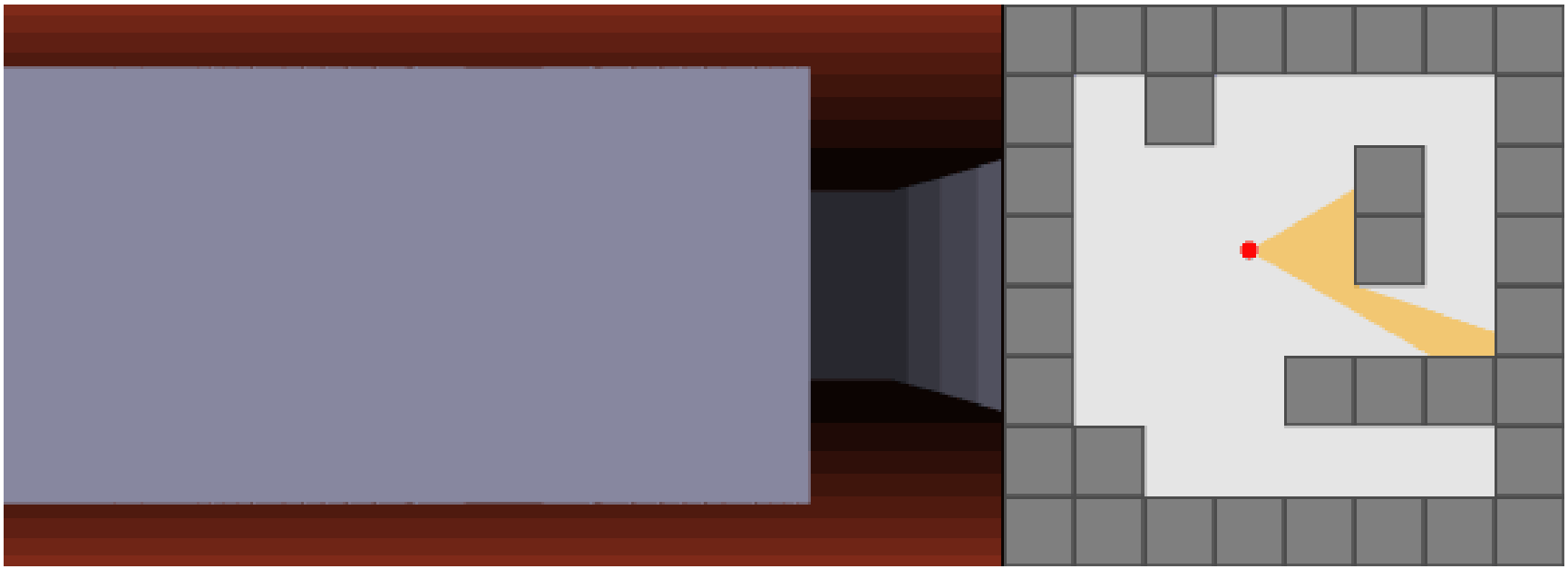
Создание кроссплатформеного псевдотрёхмерного движка (англ. engine) для 3D визуализации помещений и маршрутов в них по 2D плану.

Задачи исследования

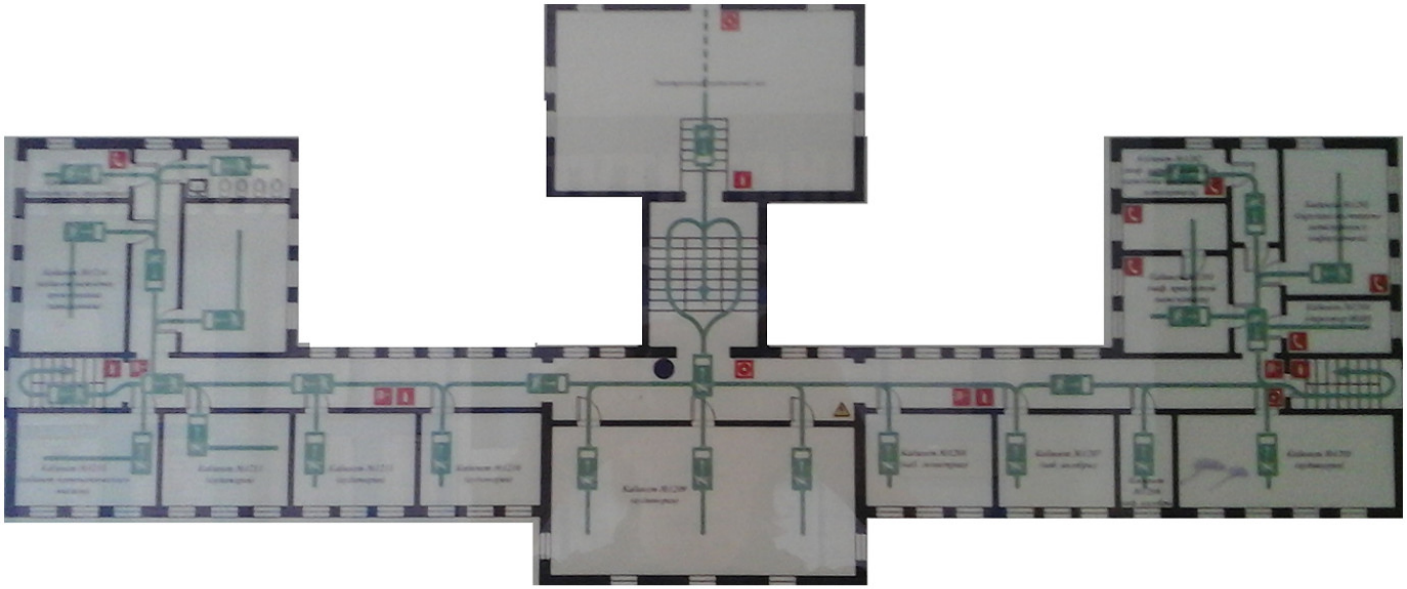
- исследование и непосредственная реализация эффективного алгоритма отрисовки проекции трёхмерной сцены
- модификация алгоритма рейкастинга для вещественных координат
- обеспечение интерактивности и поиск маршрутов
- платформа для возможной реализации дополнительных сервисов и кроссплатформености

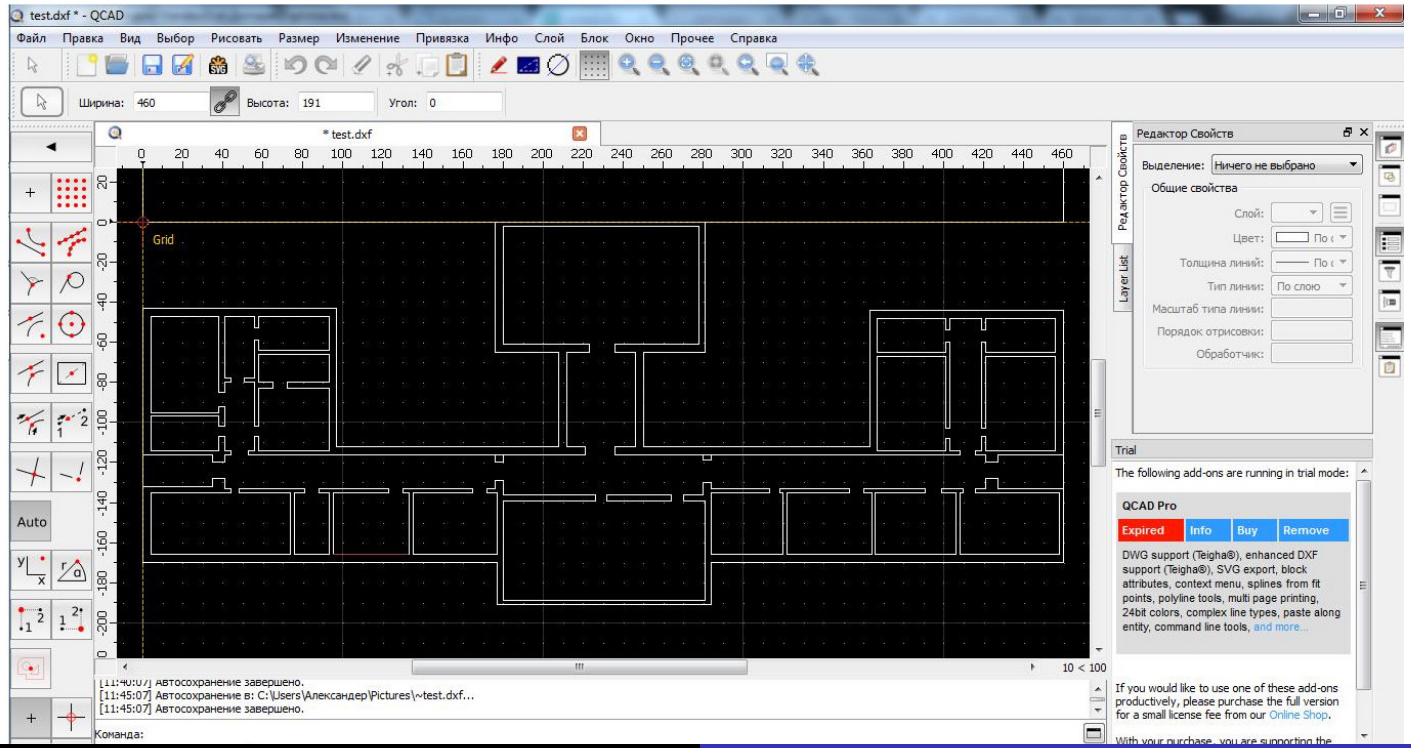
Метод бросания лучей

Метод бросания лучей (англ. raycasting, рейкастинг) - один из методов рендеринга в компьютерной графике, при котором сцена строится на основе замеров пересечения лучей с визуализируемой поверхностью.

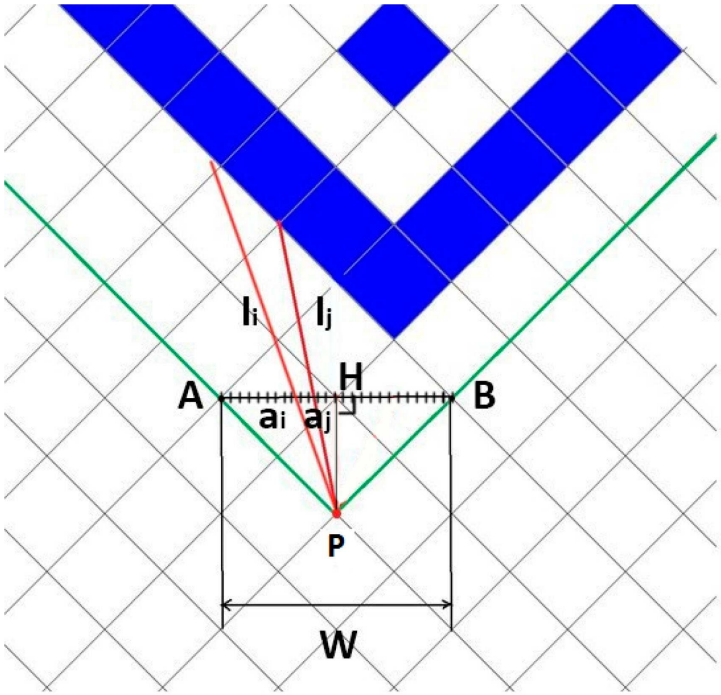


2D вид карты

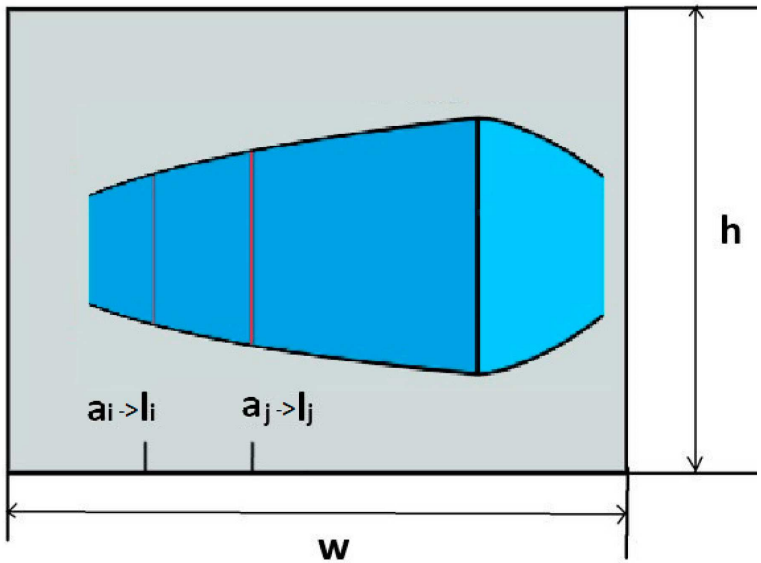



test.dxf * - QCAD
Файл Правка Вид Выбор Рисовать Размер Изменение Привязка Инфо Слой Блок Окно Прочее Справка
Ширина: 460 Высота: 191 Угол: 0
* test.dxf
Grid
Редактор Свойств
Выделение: Ничего не выбрано
Общие свойства
Слой:
Цвет:
Толщина линий:
Тип линии: По слою
Масштаб типа линии:
Порядок отрисовки:
Обработчик:
Layer List
Trial
The following add-ons are running in trial mode:
QCAD Pro
Expired Info Buy Remove
DWG support (Teigha®), enhanced DXF support (Teigha®), SVG export, block attributes, context menu, splines from fit points, polyline tools, multi page printing, 24bit colors, complex line types, paste along entity, command line tools, and more...
If you would like to use one of these add-ons productively, please purchase the full version for a small license fee from our Online Shop.
[11:45:07] Автосохранение завершено.
[11:45:07] Автосохранение в: C:\Users\Александер\Pictures\~test.dxf...
[11:45:07] Автосохранение завершено.
Команда:
Auto

Описание метода бросания лучей



$$\overrightarrow{\Delta a} = \frac{\left|\overrightarrow{AB}\right|}{w}$$
$$\overrightarrow{a_{i+1}} = \overrightarrow{a_i} + \overrightarrow{\Delta a}$$
$$\overrightarrow{a_1} = \overrightarrow{A}$$
$$\overrightarrow{a_n} = \overrightarrow{B}$$
$$\overrightarrow{r_i} = \overrightarrow{a_i} - \overrightarrow{P}$$

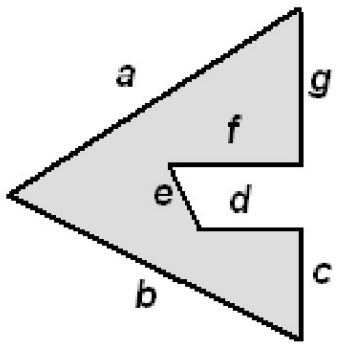


Алгоритм рейкастинга

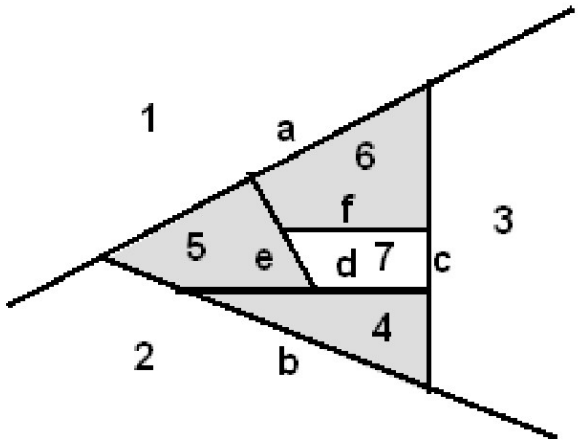
для каждой $i \in [1, n]$*:*

$l \leftarrow$ *расстояние-до-стены*$(\overrightarrow{P}, \overrightarrow{r_i})$
$h \leftarrow$ *высота-отрезка*(l)
отобразить-отрезок(i, h)

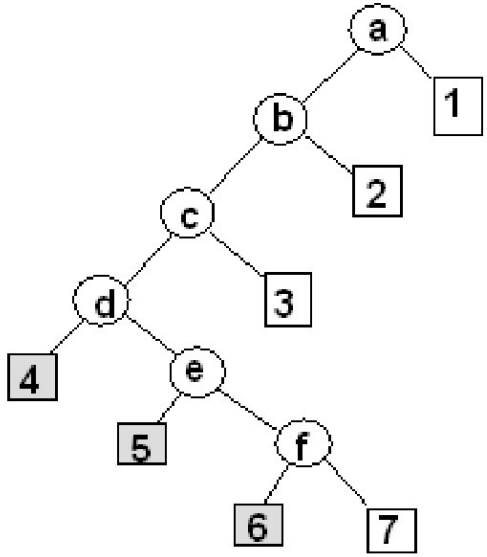
BSP-дерево



Плоское пространство

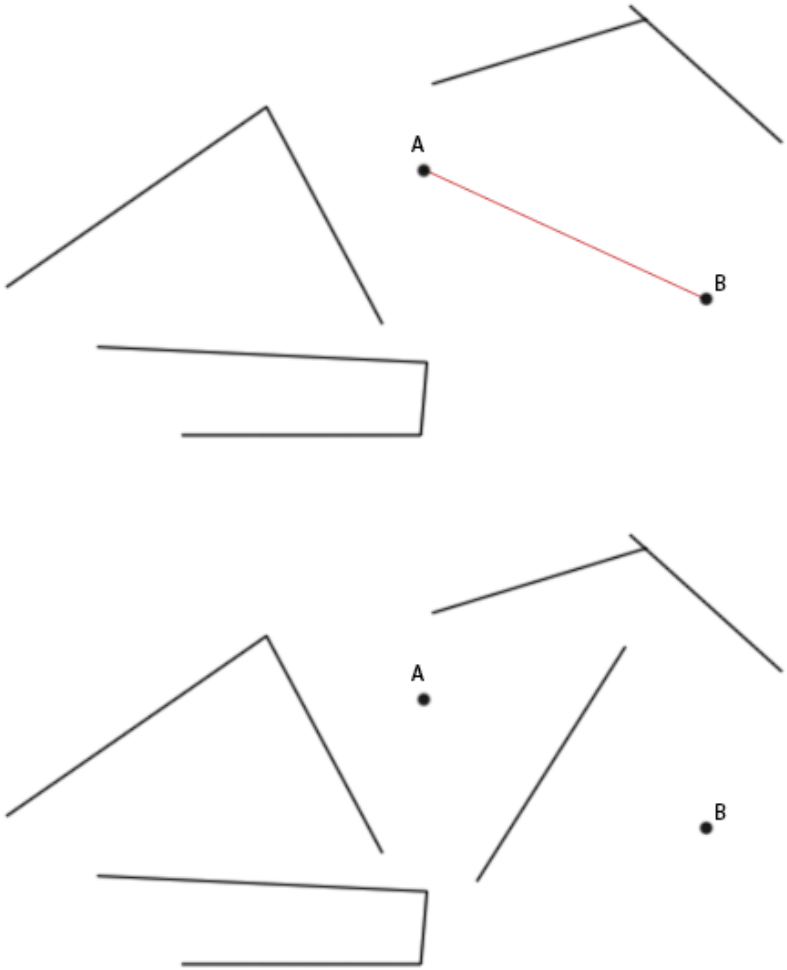


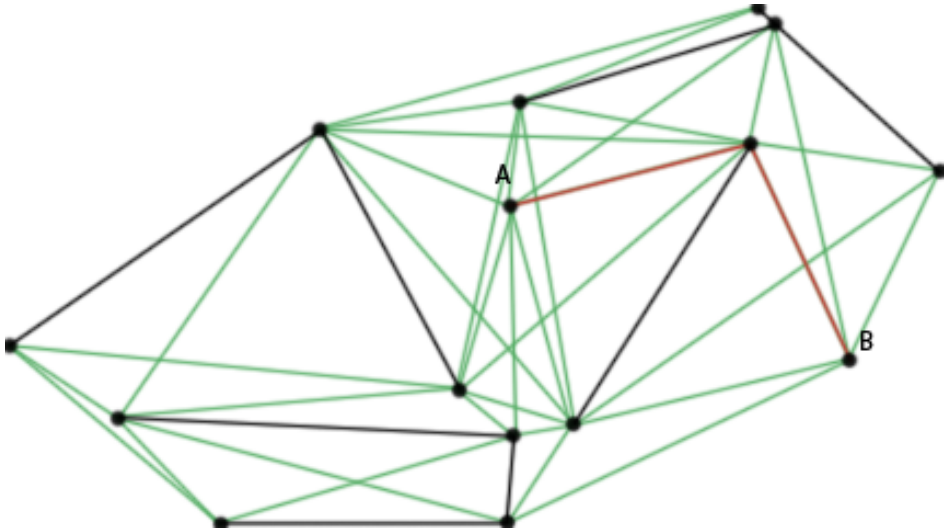
Двоичное разбиение пространства



Двоичное дерево

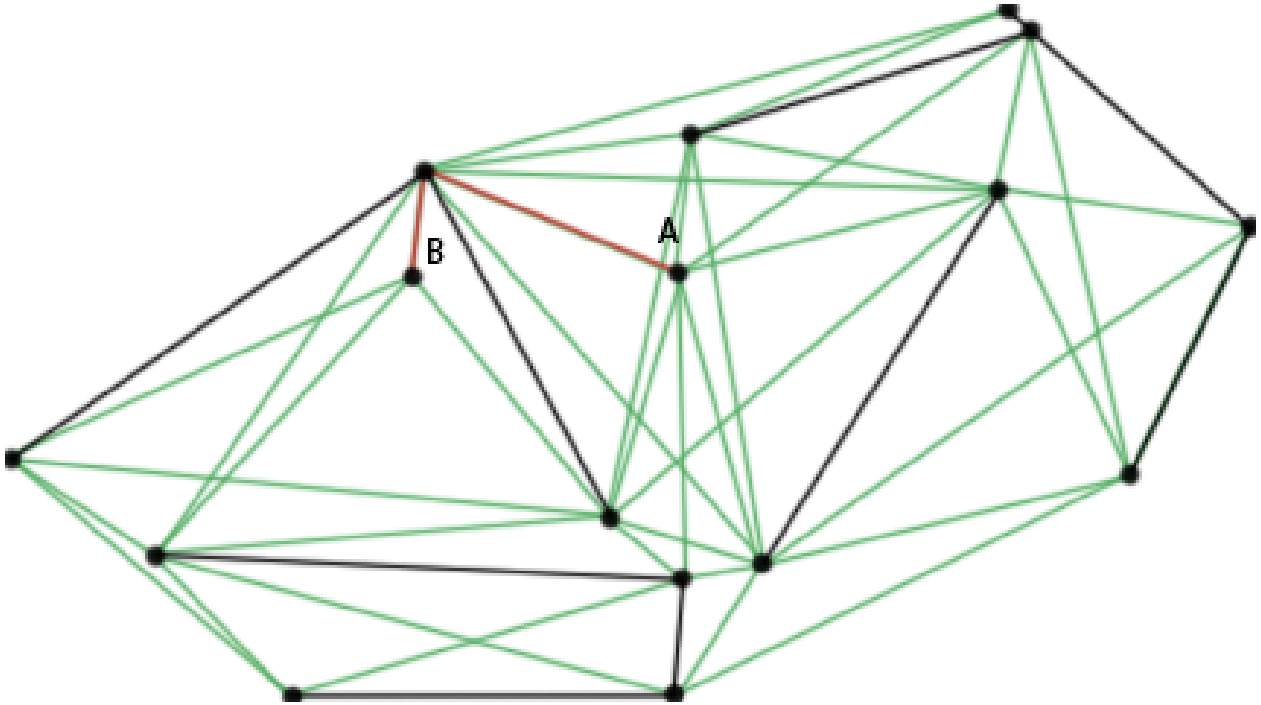
Поиск маршрута



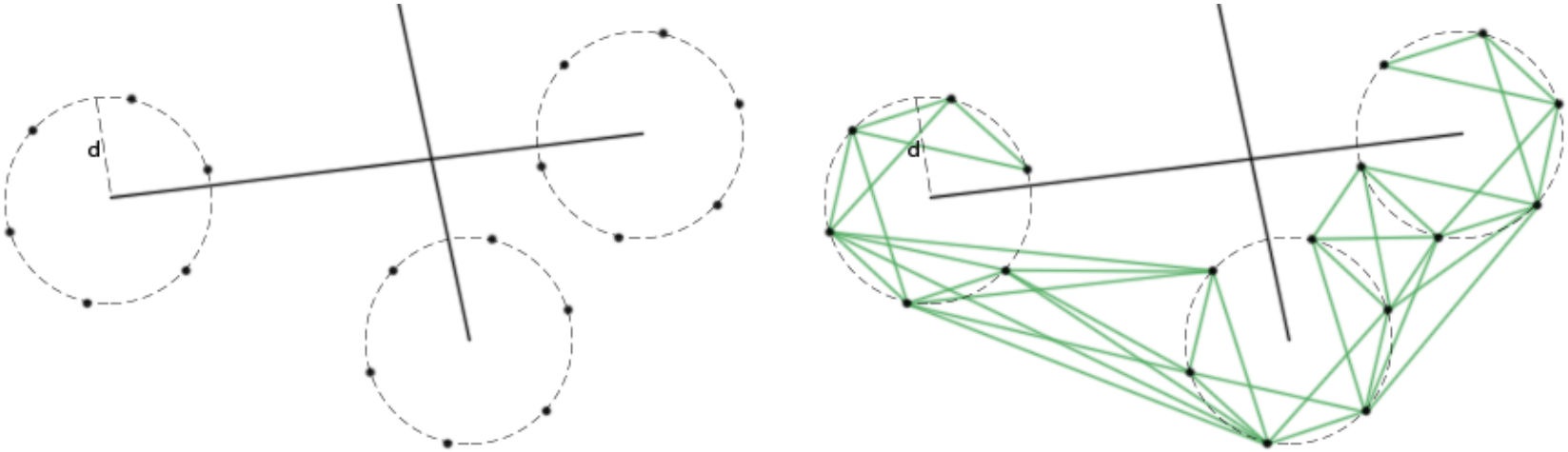


Принцип построения графа

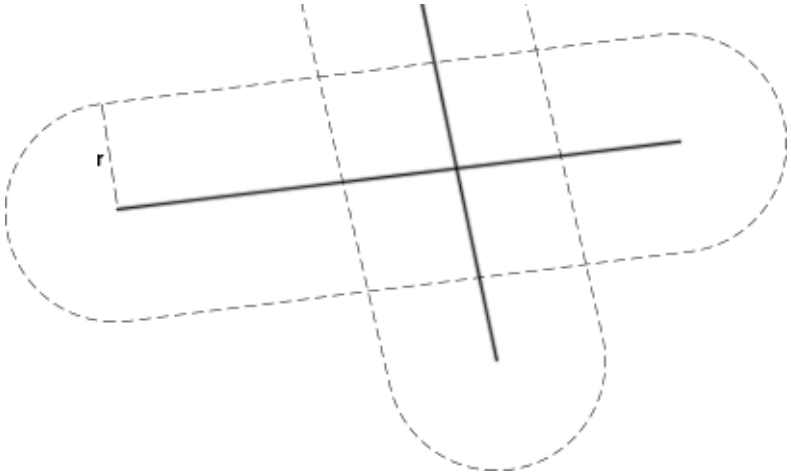
Поиск маршрута

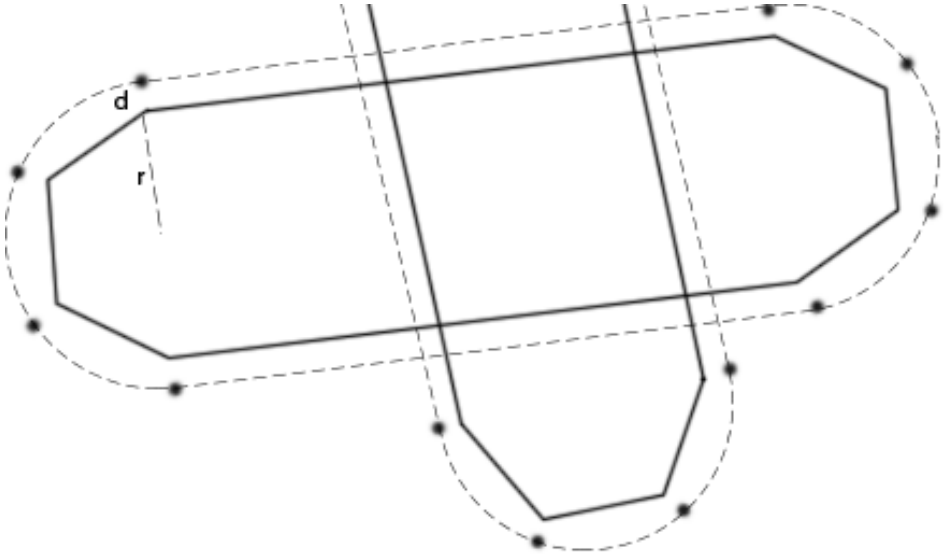


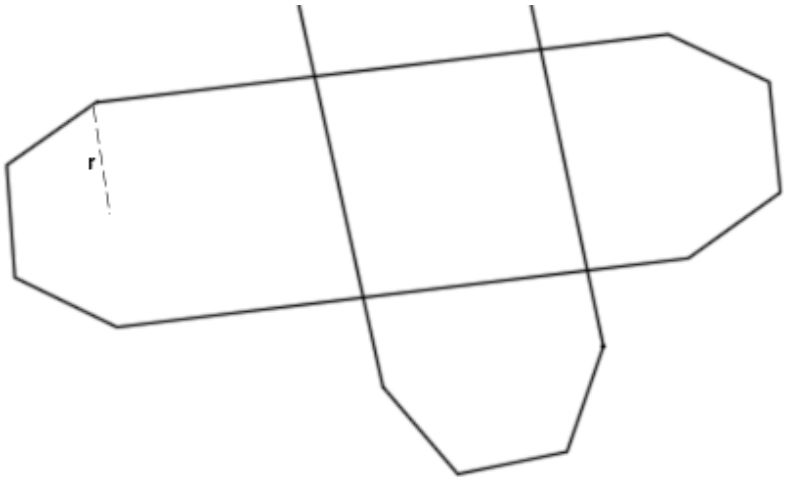
Проскакивания через углы стен

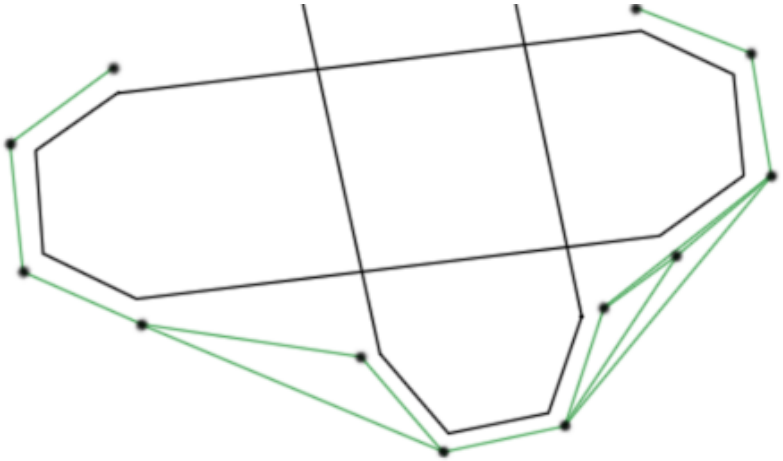


Построение вершин графа вокруг концов отрезков









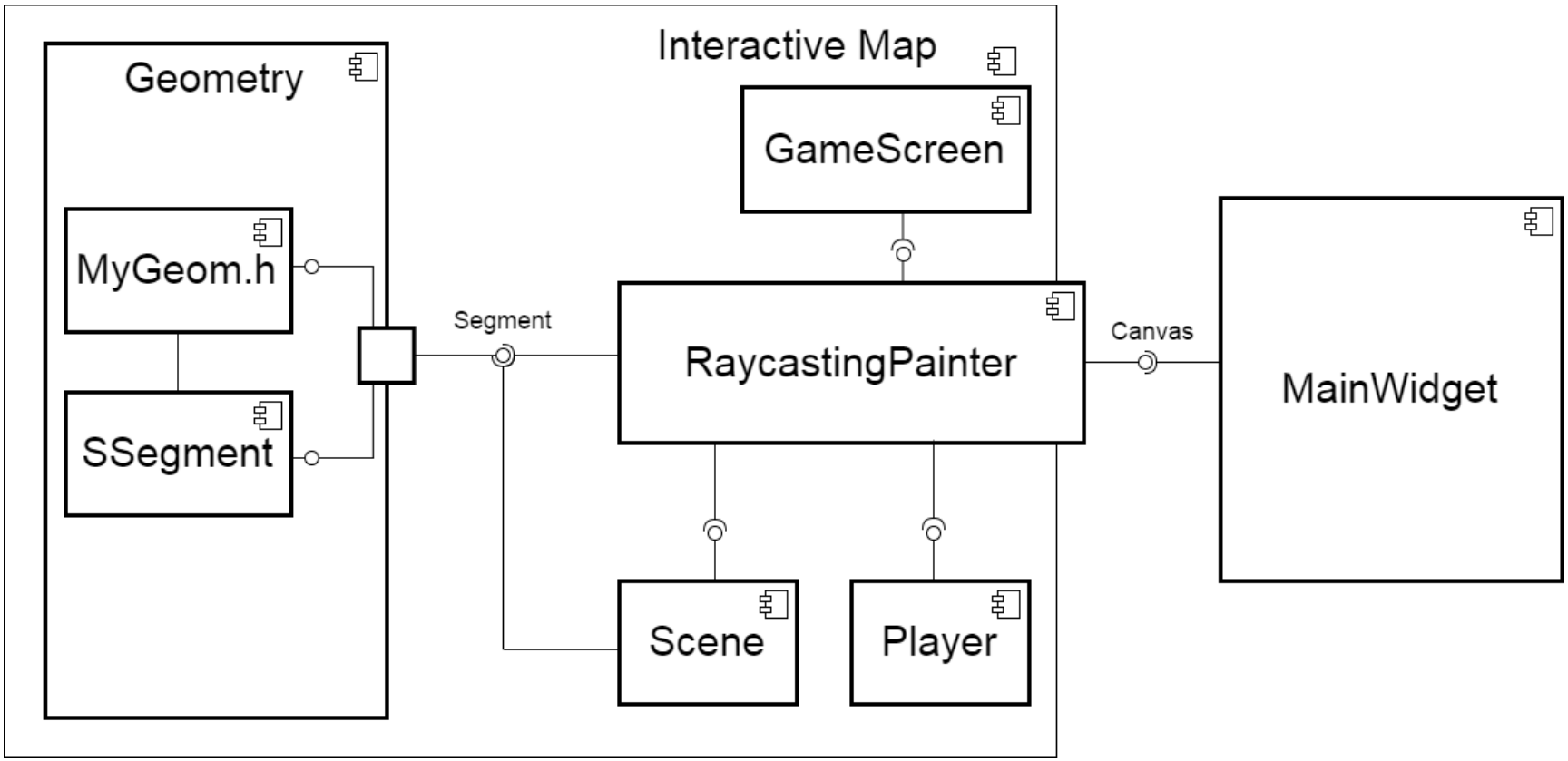
Преобразования для случая $r > 0$

Построение графа для случая $r > 0$

Алгоритм построения графа для поисков маршрута на карте

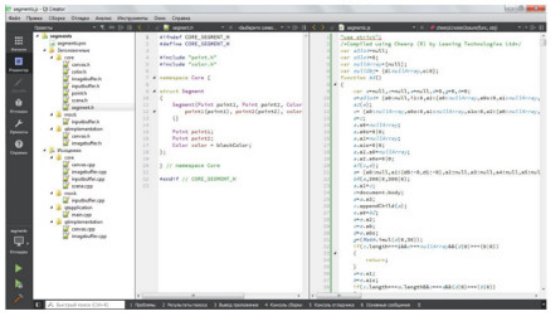
1) Для каждой точки, являющейся концом отрезка из S найти k точек равномерно расположенных на окружности радиуса $r + d$ с в этой точке. Добавить полученные точки в V.
2) Для каждого отрезка для расстояния r найти эквидистанту, приблизить её многоугольником. Удалить из S начальные отрезки и добавить отрезки, являющиеся сторонами полученных многоугольников.
3) Удалить из V вершины, лежащие внутри полученных многоугольников.
4) Для каждой пары вершин из V, рассмотреть отрезок их соеденяющий. Если этот отрезок не пересекает никакой отрезок из S, то добавить ребро между этими вершинами в E.

Диаграмма компонентов UML



Кросс-компиляция

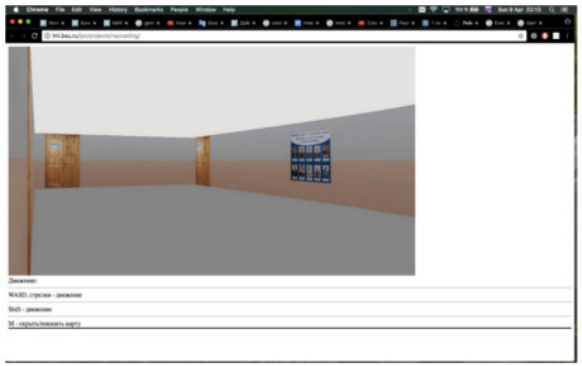
C++ Project



Algo Raycasting
View Map
Abstract Canvas
Abstract Image Buffer

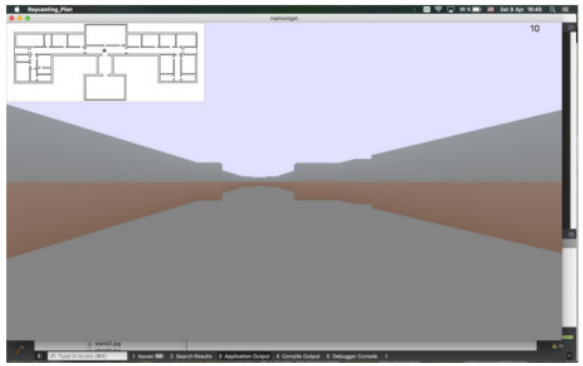
Emscripten
Web

EmscriptenCanvas(On Web)
EmscriptenImageBuffer(On Web)



QtCanvas(On Desktop)
QtImageBuffer(On Desktop)

Qt
Desktop



Результаты работы

- Модифицирован алгоритм рейкастинга для работы в вещественных координатах
- Создан движок на основе модифицированного алгоритма рейкастинга
- На основе движка сделан интерактивный план помещений корпуса ИМИ БГУ
- Проект реализован на многих платформах

Проект разрабатывается открыто, исходные коды доступны по ссылке:

https://github.com/chetca/Raycasting_Plan

Ссылка на web-реализацию:

http://imi.bsu.ru/lps/projects/raycasting/

Спасибо за внимание!

Шорников Александр Евгеньевич

3D визуализация 2D плана помещения методом бросания лучей

Таблица: Замеры производительности на ПК

OS	Memory	CPU	FPS
Windows 10	25 Mbyte	10%	27-41
Ubuntu 16.04.2 *LTS*	17.4 Mbyte	8%	34-44
FreeBSD 11.0	18 Mbyte	8%	32-47

Таблица: Замеры производительности на мобильных устройствах

Model	Antutu Test	FPS
Samsung Galaxy A5	59834	24-30
Highscreen Power Ice Evo	31672	20-26
Samsung Galaxy Star Plus	8012	12-20

Таблица: Замеры производительности в различных браузерах

Browser	Memory	FPS
Mozilla Firefox	25.4 Mbyte	42-60
Chromium	28 Mbyte	39-56
Safari	28 Mbyte	39-56
Microsoft Edge	30.6 Mbyte	34-51