



cub3D

Мой первый RayCaster с miniLibX

Описание: Этот проект вдохновлен всемирно известной одноименной игрой 90-х, которая стала первым шутером от первого лица в истории.

Это позволит вам изучить метод лучевого каста. Ваша цель будет состоять в том, чтобы

сделайте динамический вид внутри лабиринта, в котором вам придется искать свой путь.



СОДЕРЖАНИЕ

I	Предисловие	2
II	Цели	3
III	Общие инструкции	4
IV	Обязательная часть - бонусная часть	5
V	cub3D	9
VI	Примеры	11



Глава I

Предисловие

Разработан Id Программное обеспечение знаменитого Джона Кармака и Джона Ромеро, опубликованного в 1992 г. Программное обеспечение Apogee, Wolfenstein 3D это первый настоящий шутер от первого лица в истории видеоигр.



Рисунок I.1: Джон Ромеро (слева) и Джон Кармак (справа) позируют для потомков.

Вольфенштейн 3D является предком таких игр, как Doom (Программное обеспечение Id, 1993), Doom II (Программное обеспечение Id, 1994), Duke Nukem 3D (3D-мир, 1996) и Quake (программное обеспечение Id, 1996), это дополнительные вечные вехи в мире видеоигр.

Теперь ваша очередь пережить Историю заново ...

Глава II.

Цели

Цели этого проекта аналогичны целям первого года: строгость, использование С, использование базовых алгоритмов, информационное исследование и т. д.

Как проект графического дизайна, cub3D позволит вам улучшить свои навыки в следующих областях: окна, цвета, события, формы заливки и т. д.

Заключить cub3D это замечательная площадка для изучения практических приложений математики в игровой форме без необходимости разбираться в деталях.

С помощью многочисленных документов, доступных в Интернете, вы будете использовать математику как инструмент для создания элегантных и эффективных алгоритмов.



Мы рекомендуем вам протестировать оригинальную игру перед тем, как начать этот проект:

<http://users.atw.hu/wolf3d/>

Глава III.

Общие инструкции

- Ваш проект должен быть написан в соответствии с Нормой. Если у вас есть бонусные файлы / функции, они будут включены в проверку норм, и вы получите 0 если внутри есть ошибка нормы.
- Ваши функции не должны завершаться неожиданно (ошибка сегментации, ошибка шины, двойное освобождение и т. д.), За исключением неопределенного поведения. Если это произойдет, ваш проект будет считаться нефункциональным и получит 0 во время оценки.
- При необходимости все пространство памяти, выделенное кучей, должно быть правильно освобождено. Утечки недопустимы.
- Если предмет требует этого, вы должны отправить Makefile который скомпилирует ваши исходные файлы в требуемый вывод с помощью флагов - Стена, -Wextra и - Веррор, и ваш Make-файл не должен повторно связываться.
- Ваш Makefile должен как минимум содержать правила \$(ИМЯ), all, clean, fclean и ре.
- Чтобы превратить бонусы в свой проект, вы должны включить правило бонус в ваш Make-файл, который добавит все различные заголовки, библиотеки или функции, запрещенные в основной части проекта. Бонусы должны быть в другом файле _ бонус. {с / h}.
Оценка обязательной и бонусной части проводится отдельно.
- Если ваш проект позволяет вам использовать свой libft, вы должны скопировать его источники и связанные с ним Makefile в libft папка с соответствующим Make-файлом. Ваш проект Makefile должен скомпилировать библиотеку, используя ее Makefile, затем скомпилируйте проект.
- Мы рекомендуем вам создавать тестовые программы для вашего проекта, даже если эта работа **не должны быть отправлены и не будут оцениваться**. Это даст вам возможность легко проверить свою работу и работу ваших коллег. Вы найдете эти тесты особенно полезными во время защиты. Действительно, во время защиты вы можете использовать свои тесты и / или тесты партнера, которого вы оцениваете.
- Отправьте свою работу в назначенный репозиторий git. Оцениваться будет только работа в репозитории git. Если DeepThreadt назначен для оценки вашей работы, это будет сделано после ваших оценок коллег. Если во время выставления оценок Deepoughtt в каком-либо разделе вашей работы произойдет ошибка, оценка будет остановлена.

Глава IV.

Обязательная часть - cub3D

Название программы	cub3D
Сдать файлы	Все ваши файлы
Сделать файл	all, clean, fclean, re, бонус карта в формате
Аргументы	* .cub
Внешние функции.	<ul style="list-style-type: none">• открыть, закрыть, прочитать, написать, printf, malloc, бесплатно, perror, strerror, exit• Все функции математической библиотеки (-lm man man 3 math)• Все функции MinilibX
Libft авторизован	да
Описание	Вы должны создать «реалистичное» трехмерное графическое представление внутренней части лабиринта от первого лица. Вы должны создать это представление, используя принципы Ray-Casting, упомянутые ранее.

Ограничения следующие:

- Ты должен использовать miniLibX. Либо версия, доступная в операционной системе, либо из ее исходников. Если вы решите работать с источниками, вам нужно будет применить те же правила для своих libft как написано выше в Общий инструкции часть.
- Управление вашим окном должно оставаться плавным: переход на другое окно, сворачивание и т. Д.
- Отобразите различные текстуры стен (выбор за вами), которые различаются в зависимости от того, с какой стороны обращена стена (север, юг, восток, запад).

- Ваша программа должна иметь возможность отображать элемент (спрайт) вместо стены.
- Ваша программа должна иметь возможность устанавливать два разных цвета пола и потолка.
- В случае если Глубокая мысль однажды увидит ваш проект, ваша программа должна сохранить первое визуализированное изображение в BMP формат, когда его второй аргумент "--спаси".
- Если второй аргумент не указан, программа отображает изображение в окне и соблюдает следующие правила:
 - Клавиши со стрелками влево и вправо на клавиатуре должны позволять вам смотреть в лабиринт влево и вправо.
 - Клавиши W, A, S и D должны позволять перемещать точку обзора через лабиринт.
 - Нажатие ESC должен закрыть окно и полностью выйти из программы.
 - Щелчок по красному крестику на рамке окна должен закрыть окно и полностью выйти из программы.
 - Если заявленный размер экрана на карте больше разрешения дисплея, размер окна будет установлен в зависимости от текущего разрешения дисплея.
 - Использование изображений из minilibX настоятельно рекомендуется.
- Ваша программа должна принимать в качестве первого аргумента файл описания сцены с расширением .детеныш расширение.
 - Карта должна состоять только из 4 возможных символов: **0** для пустого места, **1** для стены, **2** для предмета и **C**, **Ю**, **В** или же **W** для начальной позиции игрока и ориентации нереста.

Это простая действительная карта:

```
111111  
100101  
102001  
1100N1  
111111
```

- Карта должна быть закрыта / окружена стенами, в противном случае программа должна возвращать ошибку.
- За исключением содержимого карты, каждый тип элемента может быть разделен одной или несколькими пустыми строками.
- За исключением содержимого карты, которое всегда должно быть последним, каждый тип элемента может быть установлен в файле в любом порядке.
- За исключением карты, каждый тип информации в элементе может быть разделен одним или несколькими пробелами.
- Карта должна быть проанализирована так, как она выглядит в файле. Пробелы - это допустимая часть карты, и вам решать. У вас должна быть возможность анализировать любую карту, если она соответствует правилам карт.

- Каждый элемент (кроме карты) в первую очередь представляет собой идентификатор типа (состоящий из одного или двух символов), за которым следует вся конкретная информация для каждого объекта в строгом порядке, например:

* Разрешение:

```
1920 1080 рэнд
```

- Идентификатор: **p**
- X размер рендеринга
- Y размер рендеринга

* Северная текстура:

```
HET ./path_to_the_north_texture
```

- Идентификатор: **HET**
- Путь к северной текстуре * южная текстура:

```
SO ./path_to_the_south_texture
```

- Идентификатор: **ТАК**
- Путь к южной текстуре * Западная текстура:

```
МЫ ./path_to_the_west_texture
```

- Идентификатор: **МЫ**
- Путь на западную текстуру * Восточная текстура:

```
EA ./path_to_the_east_texture
```

- Идентификатор: **EA**
- Путь к восточной текстуре *
- Текстура спрайта:

```
S ./path_to_the_sprite_texture
```

- Идентификатор: **S**
- Путь к текстуре спрайта * Цвет пола:

```
F 220,100,0
```

- Идентификатор: **F**
- Цвета R, G, B в диапазоне [0,255]: **0, 255, 255**

* Цвет потолка:

C 225,30,0

· Идентификатор: C

· Цвета R, G, B в диапазоне [0,255]: 0, 255, 255

- Пример обязательной части с минималистом. **детеныш** место действия:

```
1920 1080 рэнд
HET ./path_to_the_north_texture
SO ./path_to_the_south_texture
MЫ ./path_to_the_west_texture
EA ./path_to_the_east_texture

S ./path_to_the_sprite_texture
F 220,100,0
C 225,30,0

1111111111111111111111111111
10000000000110000000000001
101100000111000002000001
10010000000000000000000000000001
11111111011000001110000000000001
1000000001100000111011111111111
111101111111101110000010001
1111011111111011101010010001
11000000110101011100000010001
10002000000000001100000010001
10000000000000001101010010001
11000011101010111101110N0111
11110111 1110101 101111010001
11111111 1111111 111111111111
```

- Если в файле обнаружена какая-либо неправильная конфигурация, программа должна завершиться должным образом и вернуть «Ошибка \ п» с последующим явным сообщением об ошибке по вашему выбору.

Глава V

Бонусная часть



Бонусы будут оцениваться только в том случае, если ваша обязательная часть ИДЕАЛЬНА. Под СОВЕРШЕННЫМ мы, естественно, подразумеваем, что он должен быть полным, что он не может выйти из строя, даже в случае серьезных ошибок, таких как неправильное использование и т. д. По сути, это означает, что если ваша обязательная часть не наберет ВСЕ баллы во время оценивания, ваши бонусы будут быть полностью ИГНОРИРОВАННЫМ.

Список бонусов:

- Столкновения со стенами.
- Скайбокс.
- Текстура пола и / или потолка.
- HUD.
- Умение смотреть вверх и вниз.
- Прыгайте или приседайте.
- Эффект тени, связанный с расстоянием.
- Бар жизни.
- Больше предметов в лабиринте.
- Столкновения объектов.
- Зарабатывайте очки и / или теряя жизнь, собирая предметы / ловушки.
- Двери, которые можно открывать и закрывать.
- Секретные двери.
- Анимация выстрела из пистолета или анимированный спрайт.
- Несколько уровней.
- Звуки и музыка.
- Поверните точку обзора с помощью мыши.

- Оружие и плохие парни в бой!



Чтобы заработать все бонусные баллы, вам необходимо подтвердить как минимум 14 из них, поэтому выбирайте с умом, но будьте осторожны, чтобы не тратить зря время!



Вам разрешается использовать другие функции для завершения бонусной части, если их использование оправдано во время вашей оценки. Вы также можете изменить ожидаемый формат файла сцены в соответствии с вашими потребностями. Быть умным!

Глава VI.

Примеры



Рисунок VI.1: Wolfenstein3D Оригинальная игра с использованием RayCasting.

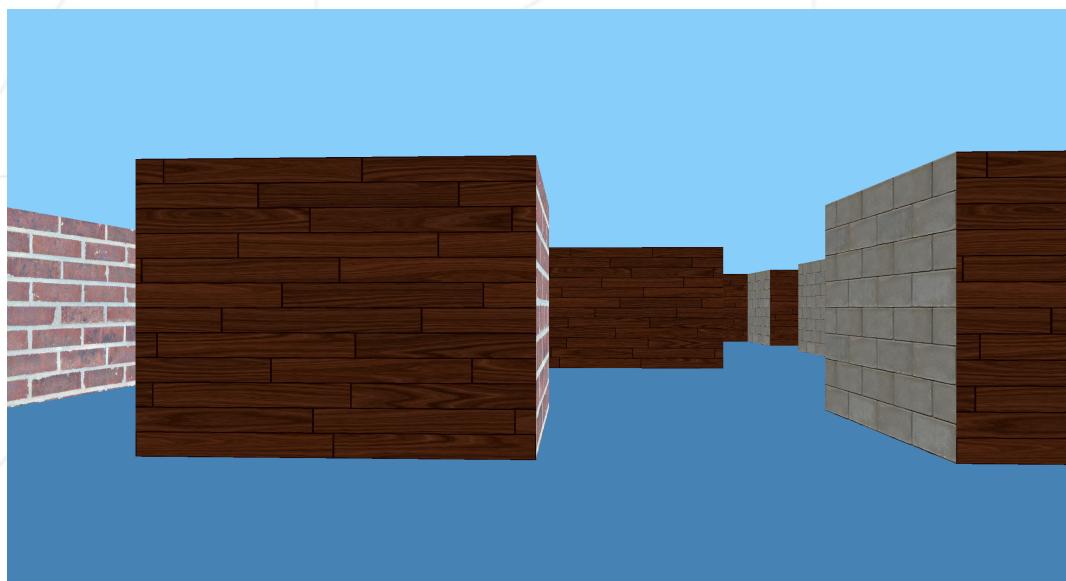


Рисунок VI.2: Пример того, как мог бы выглядеть ваш проект по обязательной части.