Мой проект

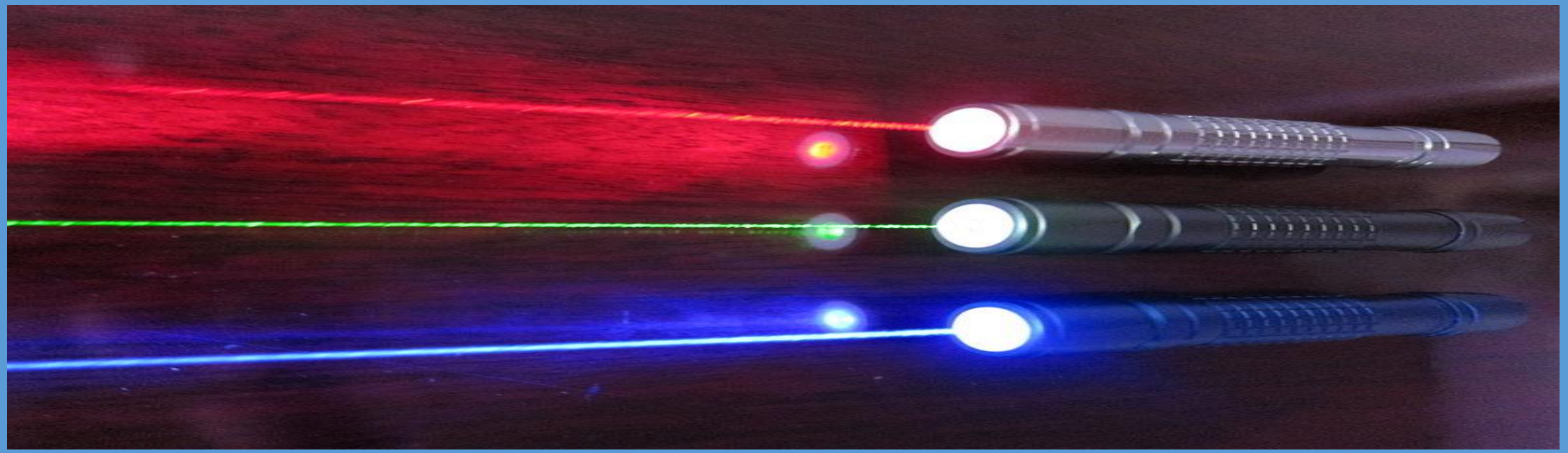
Здесь могла быть ваша реклама

О чем собственно идет речь?

Речь идет об игре, которую я придумал в голове за -1 секунду.

О чем игра?

Ты уворачиваешься от лазеров и всего что тебя убивает.



Предлагаю перейти к более информативной части

Здесь также могла быть ваша реклама

Плюсы и минусы моего проекта.

Плюсы:

Полная свобода действий;

Практически полное отсутствие задержки на девайсах разной производительности;

Редактор, в котором можно сделать свой уровень;

Многопоточность;

Своя музыка, любая;

Минусы:

Сложная постройка уровня;

Мерцание поля;

Нету рестарта;

Свобода действий, что это значит?

Свобода действий означает, что в игре вы можете сделать практически все что захотите, вы сами создатели и игроки.

Практически полное отсутствие задержки на разных девайсах

Кто-то спросит: “Откуда задержка?”, а я поясню:

На разных девайсах код работает быстрее или медленнее, что плохо сказывается на задумке автора, и только с 3 попытки у меня получилось найти способ практически полностью её устранить.

Первый способ: находить по кадрам. Очень сильно будет влиять на код, все будет становится либо слишком быстрым, либо медленным.

Второй способ: отдельный поток с таймером. Я думал, это будет идеальный способ, но у него еще больше изъянов чем у первого:

1: Он попросту считает неправильно, что очень мешает строить.

2: Он очень любит проскакивать значения.

Редактор в котором можно построить свой уровень.

Он сделан далеко не идеально, но он позволяет строить неплохие уровни.

Многопоточность

Я бы сказал что этот проект многомногопоточный, ведь каждый лазер здесь это отдельный поток, но запускается он только один раз.

```
std::thread kd_x5_y4(killing_dot_x5_y4_t, std::ref(map),couldown_lasers,laser_sleep);
kd_x5_y4.detach();
std::thread kd_x5_y2(killing_dot_x5_y2_t, std::ref(map),couldown_lasers,laser_sleep);
kd_x5_y2.detach();
std::thread kd_x5_y1(killing_dot_x5_y1_t, std::ref(map),couldown_lasers,laser_sleep);
kd_x5_y1.detach();
std::thread kd_x5_y3(killing_dot_x5_y3_t, std::ref(map),couldown_lasers,laser_sleep);
kd_x5_y3.detach();
```

Эти анимации запущены навсегда, их не надо снова подключать.

Сложная постройка уровня

Заключается она в том, что если мы хотим чтобы лазер появлялся на 20 секунде, но у нас уже был лазер на 10 секунде, то нам нужно:

20000(миллисекунды)-10000(миллисекунды)-4000(время анимаций)

```
if (i == 1)Sleep(3720);
if (i == 2)Sleep(7030);
if (i == 3)Sleep(850);
if (i == 4)Sleep(2300);
if (i == 5)Sleep(1200);
if (i == 6)Sleep(750);
if (i == 7)Sleep(2200);
if (i == 8)Sleep(2100);
if (i == 9)Sleep(3450);
```

Вопросы которые могут возникнуть

“Почему ты не создашь отдельный поток который будет запускать потоки?” – Запуск потока в потоке ломает мою игру, она перестает выводить что-либо на экран.

“Почему ты используешь “Sleep()”?” – К этому мы перейдем чуть позже.

Мерцание поля

В моей игре присутствует мерцание, ведь это не просто игра, а программа “АНТИ-ЭПИЛЕПТИК 3000”!

Мерцание происходит из-за частого обновления поля, на разных компьютерах будет по разному.

Вопросы

“В твоей игре слишком много потоков, она из-за этого и лагает!”

Хочу вас разогорчить, но даже пустое поле 3 на 3 без единого потока будет мерцать.

“Используй “\r”!” – Мы имеем дело с двумерным массивом, а не одномерным, поэтому “\r” бесполезен.

Нету рестарта

“Почему?” спросите вы, а я отвечу – мне было лень. И вот я серьезно.

Sleep – как вершина эволюции

Все еще можно купить рекламу

Почему это он - вершина эволюции?

Это ИДЕАЛЬНЫЙ способ сделать игру без задержки, у меня даже с хромом, в котором 100 вкладок, и запущенным калькулятором, стимом и гд, все работает как часы (практически).

На нем держится все в моем коде.

Как писался код?

Код писался не быстро, но все было хорошо до того момента как надо было построить уровень. Из 2 минут 36 секунд мне пришлось оставить 1 минуту 7 секунд, и это все равно было максимально долго.

Создание уровня

Это не то, как я делал уровень это пособие для **ВАС:**

1 этап – подумайте, надо ли оно вам.

2 этап – помолитесь, это должно снизить коэффициент божьей кары во время постройки уровня.

3 этап – смиритесь, возьмите калькулятор, он вам поможет.

4 этап – выберите музыку.

5 этап – переведите музыку в .wav файл.

6 этап – поместите музыку в папку с .exe файлом.

7 этап – SUFFERING BEGINNING!!!

Примечания

1: Уровень не факт что возможен, ведь он создавался в цели *демонстрации функционала программы*.

2: Если у вас не включается музыка, проверьте, в одной папке ли музыка с .exe файлом, и проверьте, чтобы путь до .exe файла не составлял больше 256 символов.

Отдельные благодарности

Говорю огромное спасибо Виктору, который помогал мне найти нужные темы.

KOZEL PRODACTION за финансирование проекта.

Мне за то, что я написал такой код.

Всем, кто посмотрел данную презентацию до конца.

THANKS FOR THANKS

Скидка 30% на рекламу