Пояснительная записка  
к курсовому проекту по дисциплине «Программирование».  
Тема проекта «Черепашья графика».

Автор: студент гр. 9373 Савинов А.О.

Цель курсового проекта – продемонстрировать умение работать с файлами и структурами данных на примере двусвязного списка. Тема - реализация концепции “Черепашья графика”. Черепашья графика — один из методов рисования графических примитивов, основанный на метафоре рисующего устройства (черепахи), которое перемещается по экрану и поворачивается в заданных направлениях, при этом оставляя (или, по выбору, не оставляя) за собой нарисованный след заданного цвета и ширины. Так же, это способ задания команд такому устройству.

В своём проекте я реализовал запись в файл типа “bmp” изображения, полученного в результате обработки программой команд для черепахи. Для того, чтобы “отдавать приказы” черепахе, мною был придуман и реализован язык черепашьей графики, который включает в себя команды по определению холста для рисования, команды по определению писчих перьев, команды черепахи и комментарии. Саму программу я реализовал в виде утилиты командной строки.

Типичный сценарий использования программы выглядит так:

1. Пользователь создаёт файл и записывает в него команды на языке черепашьей графики.
2. Пользователь запускает утилиту с параметрами: имя файла, созданного на предыдущем этапе, и необязательный параметр - имя файла, для результирующего изображения. Если второй параметр опущен, итоговое имя выходного файла будет получено из входного добавлением суффикса “.bmp”. Если во втором параметре нет суффикса “.bmp”, он будет добавлен.
3. Пользователь использует результат из файла “.bmp”.

Структуры данных и алгоритмы, реализованные в программе:

Для построения прямых линий использован алгоритм Брезенхэма:

<https://ru.wikibooks.org/wiki/%D0%A0%D0%B5%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D0%B8_%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D0%BE%D0%B2/%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC_%D0%91%D1%80%D0%B5%D0%B7%D0%B5%D0%BD%D1%85%D1%8D%D0%BC%D0%B0#%D0%A0%D0%B5%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F_%D0%BD%D0%B0_C++>

Аналогичный алгоритм использован для построения окружностей:

<http://atarasevich.blogspot.com/2015/07/on-circle-rasterization-algorithms.html>

В программе были реализованы структуры данных “двусвязный список” (DS::list), “строка” (DS::string), которые являются заменой стандартных контейнеров, и реализованы в учебных целях. На основе двусвязного списка был реализован рекурсивный алгоритм обхода списка с вложенными подсписками. В программе использовались паттерны проектирования Composite (Компоновщик) для хранения и обработки дерева команд, Simple Factory (Простая фабрика) для генерации классов команд и Decorator (Декоратор) для организации поведения классов команд.

Приложение 1. Язык черепашьей графики.

Типы данных:

n unsigned integer

xN, yN unsigned integer

a double

x, y double

x1, y1 double

x2, y2 double

Объявление цвета:

RGB n n n

Команды:

M a - move a (подвинуться на расстояние a)

MT x y - move to x,y (передвинуться в точку x,y)

TL - turn left (повернуть налево на 90°)

TR - turn right (повернуть направо на 90°)

TA a - turn clockwise a (повернуть на a градусов по часовой стрелке)

TS, TN, TW, TE - turn south, north, west, east (повернуть на юг, север, запад, восток соответственно)

PD - pen down (опустить перо - при передвижении след будет оставаться)

PU - pen up (поднять перо - при передвижении след не будет оставаться)

SELPEN n - select pen n (выбрать перо с номером n - из указанных ранее)

DO n ... AGAIN - цикл (повторить последовательность команд n раз)

Определения:

CANVAS xN yN color [VIEW x1 y1 x2 y2] END - определение холста (xN, yN - размер итогового изображения (в пикселях), color - определение цвета, VIEW необязательный параметр, определяет относительные координаты нижнего левого угла (x1, y1) и правого верхнего угла (x2, y2)). Перо устанавливается в центр холста и “опущено”. Команда должна быть употреблена ровно один раз.

DEFPEN n SIZE a color END - определение пера (n - номер, a - размер, color - цвет). Каждое перо можно определить ровно один раз. Перо с номером 0 (тонкое, черное) определено и выбрано по умолчанию.

/\* ... \*/ - комментарии (команды /\* и \*/ должны быть отделены пробелами от других)

Приложение 2. Примеры программ на языке черепашьей графики.

CANVAS 300 300 RGB 250 150 180 END /\* определяем холст размера 300 на 300 \*/

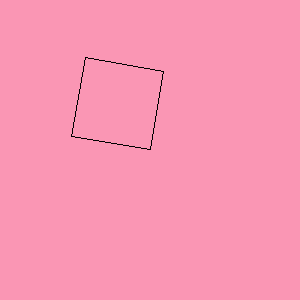
TA 10 /\* повернуть на 10° по часовой стрелке \*/

DO 4 /\* повторить 4 раза \*/

M 80 TL /\* подвинуться на расстояние 80, повернуть налево \*/

AGAIN /\* конец цикла \*/

Результат выполнения:



CANVAS 1500 1500 RGB 250 250 250 VIEW 0 0 500 500 END /\* определяем холст и относительное поле зрения \*/

DEFPEN 1 SIZE 6 RGB 255 0 0 END /\* определяем “красное” перо \*/

DEFPEN 2 SIZE 10 RGB 0 255 0 END /\* определяем “зеленое” перо \*/

PU /\* поднять перо \*/

MT 0 0 /\* передвинуться в левый нижний угол \*/

PD /\* опустить перо \*/

TA 45 /\* повернуть на 45° по часовой стрелке \*/

DO 10 /\* повторить 10 раз \*/

SELPEN 1 M 50 /\* нарисовать “зеленым” пером линию длиной 50 \*/

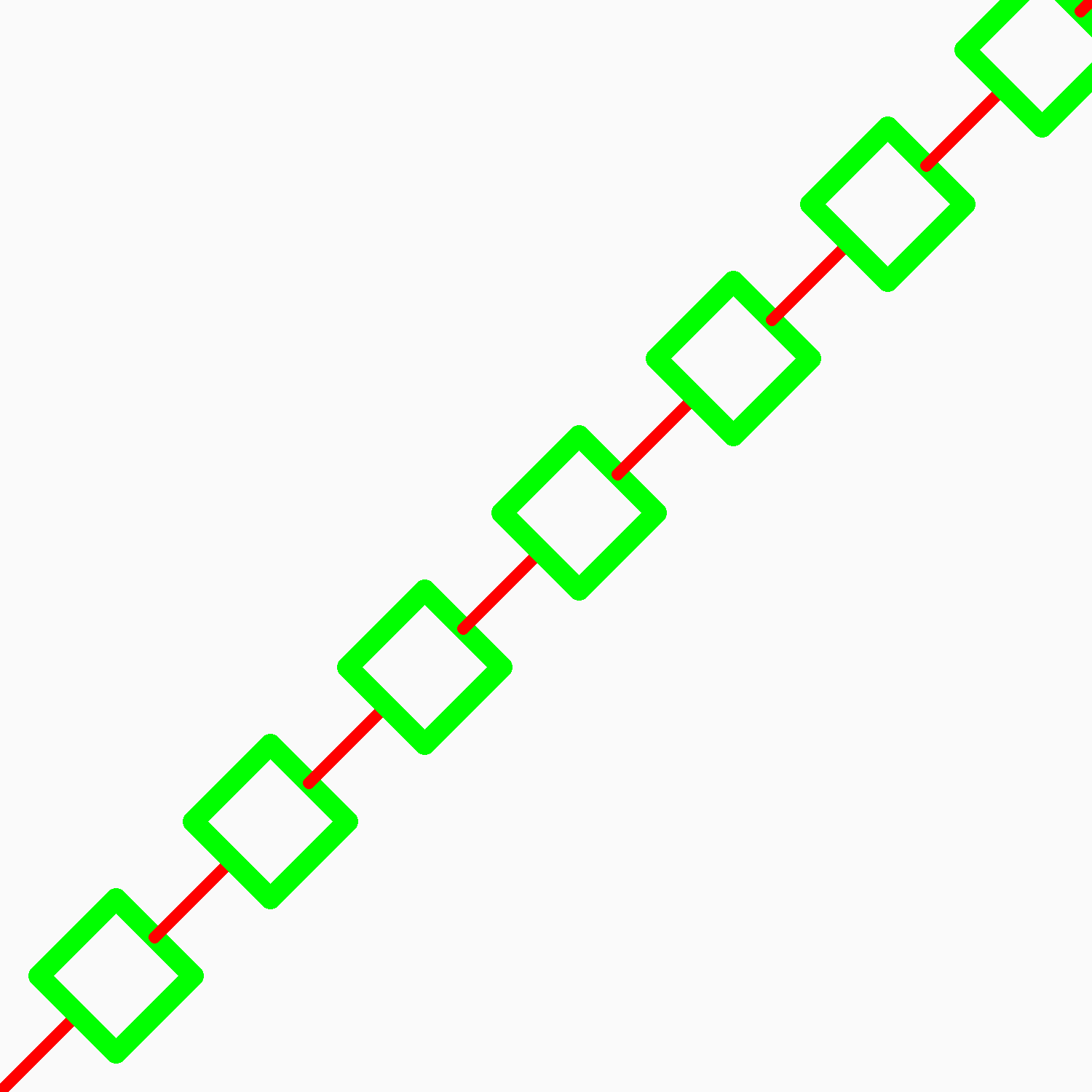
SELPEN 2 /\* выбор пера номер 2 \*/

TR M 25 DO 3 TL M 50 AGAIN TL M 25 TL /\* нарисовать “красным” пером квадрат со стороной 50 \*/

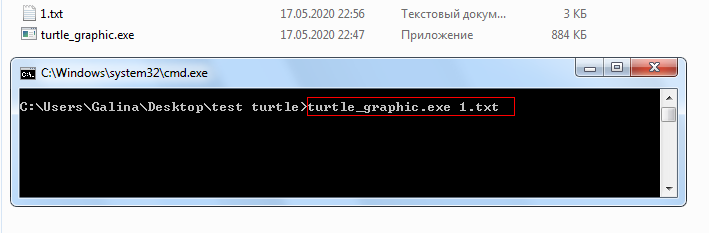
PU M 50 PD /\* переходим к следующему элементу \*/

AGAIN

Результат выполнения:



Приложение 3. Пример вызова программы.



Приложение 4. Видеопрезентация.

К настоящей записке прилагается видеопрезентация.

Приложение 5. Исходный текст программы.

К настоящей записке прилагается набор исходных файлов программы для компиляции gcc-mingw в режиме C++14, а также файл проекта для CodeBlocks. Дополнительно приложено несколько программ для черепахи.