Пояснительная записка для проекта “Черепашья графика”

Тема курсового курсового проекта - 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. А так же способ задания команд такому устройству.

В своём проекте я реализовал возможность записывать в файл с расширением “.bmp” изображения, полученные в результате работы программы. Также, для того, чтобы “отдавать приказы” черепахе, мною был придуман и реализован язык черепашьей графики, который включает в себя команды по определению холста, команды по определению писчих перьев, команды черепахи и комментарии. Саму программу я реализовал в качестве консольной утилиты.

Одна итерация по созданию итогового изображения черепашьей графики выглядит так: в начале пользователь создаёт файл и записывает в него команды на языке черепашьей графики. После этого он запускает утилиту с параметрами: имя файла с командами и имя файла, куда нужно поместить изображение(без “.bmp”). Если в качестве параметра введено лишь имя файла с командами, например, “1.txt”, то итоговое изображение будет помещено в файл “1.txt.bmp”. После этого пользователь запускает утилиту с введёнными параметрами. При этом, если исходный набор команд черепахи был задан корректно, то формируется изображение, а иначе выводится сообщение об ошибке.

Структуры данных и алгоритмы, реализованные в программе:

Для построения прямых 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++](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" \l "%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>

Так же в программе была реализована структуры данных “двусвязный список”(list), и “строка”(string), которые могут использоваться и в других проектах. 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, Factory Method(Фабричный метод) для генерации классов команд и Decorator(Декоратор) для организации поведения классов команд.

Дополнения:

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)

TR - turn right(повернуть направо)

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

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 и y1) и правого верхнего угла(x2, y2))

DEFPEN n SIZE a color END - определение пера(n - номер, a - размер, color - цвет)

/\* ... \*/ - комментарии(перед и после символов /\* и \*/ обязательно ставить пробелы)

2)Пример ввода одного параметра утилиты

