SVG ФАЙЛОВЕ

Тема 8

Github repository: https://github.com/keti200125/SVGfiles

За реализирането на този проект ще работим с базовите фигури в SVG. Нужно е програмата ни да поддържа команди, свързани с обработката на файловете и информацията, която съдържат. Операциите трябва да могат да се изпълняват от потребителя.

Класа BasicShapes е абстрактен. Той съдържа чисто виртуални функции. Него ще наследяват останалите класове в полиморфната йерархия. В него е вкарана библиотеката <string>, която е позволено да ползваме в рамките на този проект, за да избегнем динамичното заделяне на памет за цвета и съответно реализилането на голяма четворка.

Класовете, които наследяват класа BasicShapes са съответните фигури, които избрах да реализирам: Circle, Rectangle, Ellipse, Line. Всяка от тях има функция за подаване на информация на поток, също функции clone и print, които ще са ни полезни в следващата част на реализацията.

BasicShapesCollection е клас, който е колекция от фигурите, които сме създали. Благодарение на него имаме възможност да правим промени(създаване на нови фигури, изтриване на някои) без да е нужно незабавно да бъдат включени във файл. Избрах да го реализирам без <vector> ,за да се упражня в създаването на колекця с динамичо заделяне на памет. Тук отново имаме полезни функции, които ще са ни необходими в следващия клас.

writeInFile(std::ostream& os)-подаваме цялата информация за всички фигури в колекцията в поток

freeAllSpace()-премахваме всички елементи от колекцията

printAll()-принтираме цялата информация за фигурите на конзолота

removeShapeByIndex(std::size\_t index)-премахваме дадена фигура по подадено място, което се намира и ако липсва такава на съответната позиция,изкарваме съответното съобщение

addRectangle(int x, int y, std::string fill, int width, int height);

addCircle(int x, int y, std::string fill,int radius)

addLine(int x1, int y1, std::string fill, int x2, int y2)

addEllipse(int x, int y, int rX, int rY, std::string fill)

и addShape(BasicShapes\* basicShape) са функции, които използваме за добавянето на елементи.

В клас Command реализирам командите и стартирането на програмата от конзолата.

Може би не е лоша идея зареждането на програмата да се направи в отделен клас или функциите за командите да са private.Аз избрах да ги реализирам заедно.

open(std::string fileName) –в първата команда отваряме файл или създаваме, ако такъв липсва. В първия случай трябва да извлечем информацията от файла и да я разпределим в колекцията.За да го постигна първо намирам размера на файла. После копирам данните в масив от char-ове.След което по указания затваряме файла. Знаем вида, в който трябва да бъдат формите, които търсим.В цикъл, който приключва, когато минем целия масив, слагаме различните случаи. Пример за кръг:

<circle cx="5" cy="5" r="10" fill="blue" />

Основото, което ползвам тук е функцията readNumbеr(), с помощта на която каствам от char към int. Бихме спестили писането на толкова код с избора на двоични файлове, защото по този начин може директно да досъпим информацията, която ни е нужна. В този слечай след като тряба да работим в svg формат предпочетох да мога да разчитам информацията, която се появава във файла, за да си правя проверка.

close()-затваря файла и изтрива направените промени, ако не са били направени в текущия файл.(въпреки това остават запазени допълнителния файл recovery)

help()-потребителят получава информация за командите

print()-принтираме информация за формите

savе()-запазваме информацията в текущия файл

saveAs(std::string fileName)-запазваме ниформацията в друг файл, посочен от потребителя

exit(std::string fileName)- функция за изход, потребителят може да реши дали желае да запази промените, или не

erase(size\_t n)-изтриваме форма на определена позиция

create()-Извличаме информация от потребителя. Реализирах тази функционалност изцяло във функцията за стартиране на програмата

Функцията within() не успях да завърша сега, но имам идеи за реализирането и. Начинът за извикване е разписан в run();

run()-Това е функцията за стартиране на програмата. Докато не се избере exit(), потребителят може да въвежда различните команди. Като първо прочитаме командата и след това отделно останалата информация.