Проект по Обектно Ориентирано Програмиране

XML парсер

Изготвил: Благовест, група 8, 7MI0800003

1. **Увод**
   1. **Описание и идея на проекта**

Проектът представлява конзолна програма, която чрез команди от потребителския вход манипулира XML файлове.

* 1. **Цел и задачи на разработката**

Целта на проекта е чрез ООП логика да бъде изградено конзолно приложение, което поддържа въвеждане на потребителски команди в диалогов режим.

Поддържаните операции по условие са следните:

* Опериране с файловете (отваряне, затваряне, запазване, запазване като)
* Принтиране съдържанието на XML файла
* Селектиране на атрибут
* Присвояване на стойност на атрибут
* Разглеждане на n-тия наследник
* Изтриване на атрибут
* Добавяне на нов наследник
* XPath заявки:

Оператор /

Оператор []

Оператор @

Оператор =

1. **Проектиране**
   1. **Обща архитектура – ООП дизайн**

Реализацията на проекта е постигната чрез:

* Базов клас Node, наследяван от двата типа елементи – TextNode (текстов елемент) и ComplexNode (елемент с вложени деца).
* Командите, които се извършват на ниво операции с файлове се извикват от класа FileManager (Singleton)
* Командите, боравещи със съдържанието на файловете – CommandManager.
* Всичко се управлява от Engine (Singleton) класа

За целите на проекта са разработени няколко темплейтни структури от данни, а именно:

* Dictionary – речник с ключове и стойности към тях
* Pair – наредена двойка, използвана в Dictionary
* Vector – стандартен динамичен преоразмеряващ се масив
* PVector – динамичен, преоразмеряващ се масив за указатели (по-специфично за Node\*)

*Node* е базовият клас, който съдържа следните атрибути:

    std::string type; - тип (текстов или родителен)

    std::string id; - идентификатор

    std::string name; - име

   unsigned level; - ниво на вложеност във файла

    Dictionary<std::string, std::string> attributes; - атрибути

Node\* parent; - родител

TextNode надгражда тези атрибути със:

std::string text; - вложен текст

ComplexNode надгражда със следния:

PVector<Node> children; - преки деца

Цялостната идея е обхождането на елементите да става рекурсивно чрез дървовидна структура от данни.

1. **Реализация, тестване**
   1. **Реализация на класове**

Клас *CommandManager* съдържа в себе си най-важните функции, а именно:

CommandManager(std::fstream&) – конструктор, извършващ парсването на файла

    void insertRoot(Node\*) – добавя кореново дърво в гората (която реално е съдържанието на целия файл)

    Node\* getNode(PVector<Node> nodes, const std::string&) const – рекурсивен гетър за елемент с конкретно id

void streamPrintAll(std::ostream&) const – използване на поток за отпечатване на съдържанието на XML дървото на общо основание

    void print() const – използване на потока std::cout за отпечатване на дървото в конзолата

    void select(const std::string& id, const std::string& key) const;

    void set(const std::string& id, const std::string& key, const std::string& value);

Тези два метода се занимават със заявките по намиране или заменяне на стойност на атрибут на конкретен елемент.

* 1. **Управление на паметта и алгоритми. Оптимизации.**

Управлението на паметта на базово ниво и в двата класа се изпълнява от методите *copyFrom(const \_\_&)*  и *free()*.

Използва се резлизация на хетерогенен контейнер с виртуален копи конструктор clone().

Използва се дървовидна структура и указатели за увеличаване на ефективността на търсенето и правенето на промени по файловете.

За изчистване на диалоговия прозорец се използва функцията *system(“clear”)*, взета от *<stdlib.h>*.

1. **Заключение и бъдещи подобрения**

Приложението беше разработено частично. Бъдещи подобрения са главно по изпълняване на условието.

**Връзка към хранилище, съдържащо реализацията на проекта:** [Github](https://github.com/just8do8it/Computer-Science-at-FMI/tree/main/Object%20Oriented%20Programming/XML%20Parser)

--------------------------------------------------------------------------------------------------------------------

**Съдържание**

Увод………………………………………...стр. 1

Проектиране………………………………..стр. 2

Реализация, тестване………………………стр. 3

Заключение и бъдещи подобрения……….стр. 4