

1.Тема - Приложение за работа с електронни таблици.

2.Архитектура:

Приложението използва следните класове:

1. Element - виртуален клас, неговите наследници са елементи в таблицата, като насочването към тях става чрез указател от тип Element. Всички класове наследници на Element имплементират виртуалната функция `getStrValue()`, която връща елемента им преобразуван до string;

2. Integer - клас наследник на Element, представляващ цяло число. Например:

Представяне в конзолата:	Представяне във файл:
123	123
+123	123
-123	-123

3. Fraction - клас наследник на Element, представляващ дроб. Например:

Представяне в конзолата:	Представяне във файл:
123.456	123.456
+123.456	123.456
-123.456	-123.456

4. String - клас наследник на Element, представляващ низ от символи. Например:

Представяне в конзолата:	Представяне във файл:
Hello world!	"Hello world!"
C:\temp\	"C:\temp\\"
"This is a quotation"	"\"This is a quotation\""

5. EmptySpace - клас наследник на Element, играещ ролята на празна клетка.

6. Formula - клас наследник на Element, представляващ низ от символи, започващ с '=';
(за момента нефункциониращ по предназначението му);

7. Table - клас представляващ таблиците, заредени от файлове;

Член данни на класа Table:

- 1) string currFileName - името на файла, чийто информация се намира в обект от класа Table;
- 2) fstream file - потока на файл чрез който извличаме и вкарваме информация;
- 3) vector<vector<Element*> > elements - двумерен масив от вектори, съдържащ информацията извлечена от файловете;

Член функции на класа Table:

- 1) bool open - зарежда информацията от файл по зададен път към файла;
 - ако не съществува файл със зададения път се създава нов;
 - ако файла е отворен връща **вярно**;

-ако файла не е отворен връща **грешно**;

2) void close - ако има отворен файл го затваря и изтрива информацията от масива elements;

3) bool save - запазва информацията от масива elements във файла от който е била заредена информацията първоначално;

-ако запазването е успешно връща **вярно**;

-ако е възникнала грешка по време на запазването връща **грешно**;

4) bool saveAs - запазва информацията от масива elements във файл, на който му се задава пътя;

-ако не съществува файл с този път се създава нов и информацията се записва в него;

-ако запазването е успешно връща **вярно**;

-ако е възникнала грешка по време на запазването връща **грешно**;

5) void print - извежда на конзолата цялата информация запазена в масива elements, представена като завършена таблица;

6) bool edit - по зададени параметри за ред, колона и нова стойност, променя предишната стойност на клетката, намираща се на тази позиция, а ако не съществува такава клетка, създава нова със зададената стойност;

помощна функция: 7) Element* factory - по зададен низ (input) преценява най-подходящия тип, на който съответства низа и спрямо този тип заделя нов елемент от този тип (типът съответства на някой от наследниците на Element) и връща указател към него от тип Element*;

3. Потребителски интерфейс:

1. open <file name> - където <file name> е пътя на даден файл; зарежда информацията от файла в масива elements на обект от тип Table; успешно отворен файл позволява използването на други команди (в противен случай другите команди не работят);

2. edit - позволява въвеждане на позиция и стойност, която да бъде зададена на нея; ако позицията не съществува се създава нова със зададената стойност;

3. print - извежда на екрана завършена таблица със стойностите на масива elements на обект от клас Table;

4. save - запазва стойностите на масива elements на обект от клас Table във файла, който е бил последно зареден;

5. saveAs <file name> - запазва стойностите на масива elements на обект от клас Table във файл по зададен път на файл (<file name>);

6. close - ако има отворен файл го затваря; изтрива елементите от масива elements;

7. exit - приключване на приложението;