

**Софийски университет „Св. Климент Охридски“
Факултет по математика и информатика**

ПРОЕКТ ЗА КУРСА ПО ОБЕКТНО-ОРИЕНТИРАНО ПРОГРАМИРАНЕ

Тема: Приложение за работа с електронни таблици

Изготвен от:
Андрей Калчев

СОФИЯ

2 0 2 3

ПЪРВА ГЛАВА

Увод

1.1. Описание и идея на проекта

Идеята на проекта "Приложение за работа с електронни таблици" е да се напише програма, която поддържа функции за обработка на данни, записани в текстови файлове. Тя трябва да може да ги прочита, валидир, принтира, редактира и запазва.

1.2. Цел и задачи на разработката

Целта на разработката е да се напише програма, която да прочита данни от текстов файл и да ги представя във вид на електронна таблица, която трябва да има основните функционалности на електронните таблици.

Първата задача е да се създаде подходяща йерархия от класовете за различните типове клетки.

Втората задача е да се напише функция, която да определя типа на клетката спрямо прочетените данни и функция за създаване на различните типове клетки.

Третата задача е създаване на клас Таблица, който да съхранява множеството от клетки.

Четвъртата задача е да се създаде клас за работа с класа Таблица и за обработка на зависиците от цялостната таблица клетки - клетките от тип формула.

Последната задача е да се направи клас, който да валидира и изпълнява различните команди за работата с таблиците, въвеждани в терминала.

1.3. Структура на документацията

Документация на проекта се състои от: Заглавна страница, Уводна страница, Глава с описание на плана за изпълнение на задачата и основни сложности, Глава описваща архитектурата на проекта, Глава за основната реализация, Заключение и Използвана литература.

ВТОРА ГЛАВА

Преглед на предметната област

2.1. Основни дефиниции, концепции и алгоритми, които ще бъдат използвани

В проекта са използвани основните концепции на обектно ориентираното програмиране: енкапсулация, абстракция, преизползване и полиморфизъм. Има отделен клас за създаване на различни инстанции на клетки.

2.2. Дефиниране на проблеми и сложност на поставената задача

Първият проблем е преобразуването на данните от текстовия файл в конкретния тип клетка, което се усложнява от проверката за изпусната запетая.

Вторият проблем са празните клетки, които не са отбелязани категорични като такива със запетайи.

Третият проблем е, че ако формулите реферират друга клетка, стойността на формулата може да се сметне само в контекста на цялата таблица.

Последния проблем е, че формулите могат да се реферират посредством реферирание на други формули.

2.3. Подходи, методи за решаване на поставените проблеми

Първият проблем се решава много функции за определяне на типа на прочетените данни.

Вторият проблем се решава с допълнителна характеристика, която се пази в класа Таблица - брой колони.

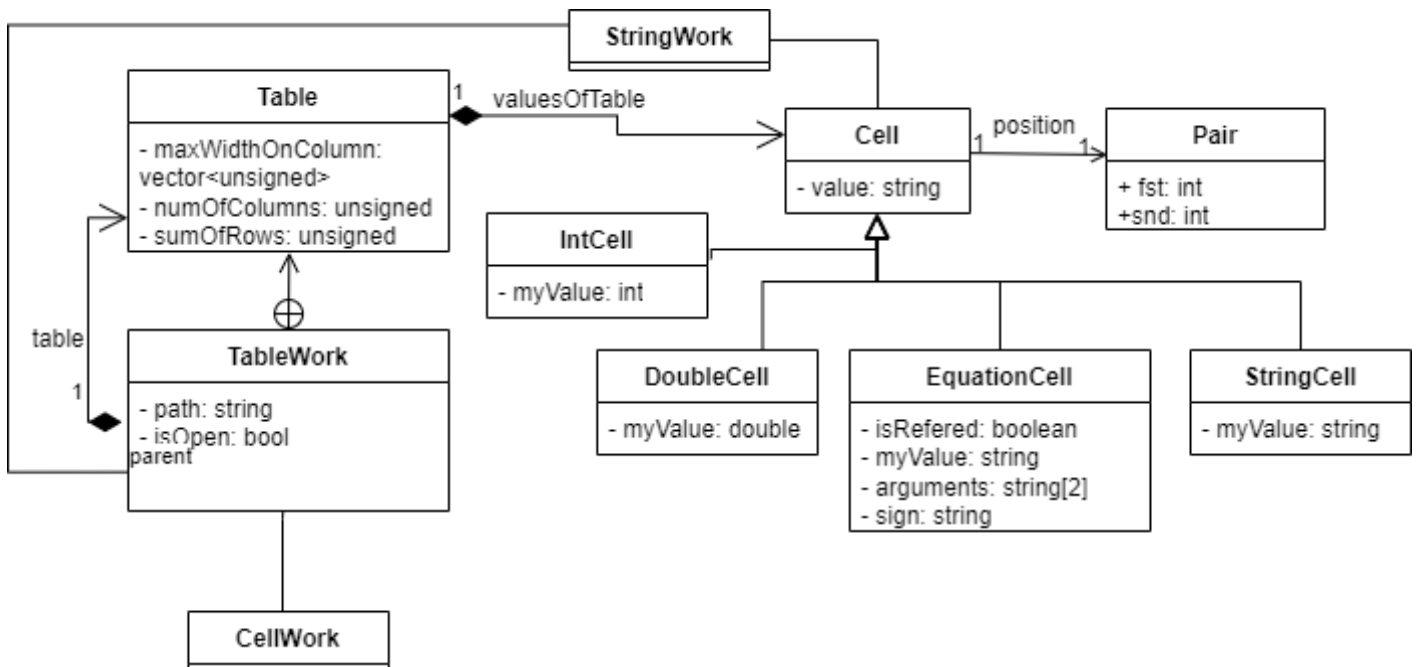
Решението на третият проблем е функциите са пресмятане на стойността на формулите да са преместени в класа за работа с таблици.

Последният проблем се решава с рекурсия.

ТРЕТА ГЛАВА

Проектиране

3.1. Обща архитектура – ООП дизайн



Фиг. 3.1 Диаграма на класовете

На фигура 3.1 е показана общата архитектура на проекта. Класът Клетка е абстрактен базов клас. Класовете, които го наследяват са клетки от тип цели числа, дробни числа, стринг и формула, всяка от която съдържа стойност от съответния тип данни. Двойка е структура написана, за да се съхраняват колоните и редовете на клетките на едно място и се използва в базовия клас Клетка и при пресмятането на формулите в Таблица. Таблица е клас, който съдържа вектор от тип указател към клетка, за съхранение на множество клетки, а класът за работа с таблици съдържа методите за манипулиране на таблици.

3.2. Важни извадки от кода

```
static Cell* createCell(unsigned row, unsigned column, std::string value)
{
    value = StringWork::removeWhiteSpaces(value);

    if(StringWork::isInt(value)){
        return new (std::nothrow) IntCell(row, column, value);
    }
    else if(StringWork::isDouble(value)){
        return new (std::nothrow) DoubleCell(row, column, value);
    }
    else if(StringWork::isString(value)){
        return new (std::nothrow) StringCell(row, column, value);
    }
    else if(StringWork::isEquation(value)){
        return new (std::nothrow) EquationCell(row, column, value);
    }

    throw std::invalid_argument(value);
}
```

Фиг. 3.2 Функцията за създаване на обект от клас Клетка

На Фиг. 3.2 е показана статичната член функция на класа за работа с клетки, която създават обект спрямо стринг, който е прочетен от файл. В тази функция се виждат и функциите от класа за работа със стрингове, за определяне на типа на стринга.

ЧЕТВЪРТА ГЛАВА

Реализация, тестване

4.1. Реализация на класове

Абстрактен базов клас Клетка - Съдържа абстрактен метод, който връща дробно число, което се използва за при смятането на стойността на клетката във формули.

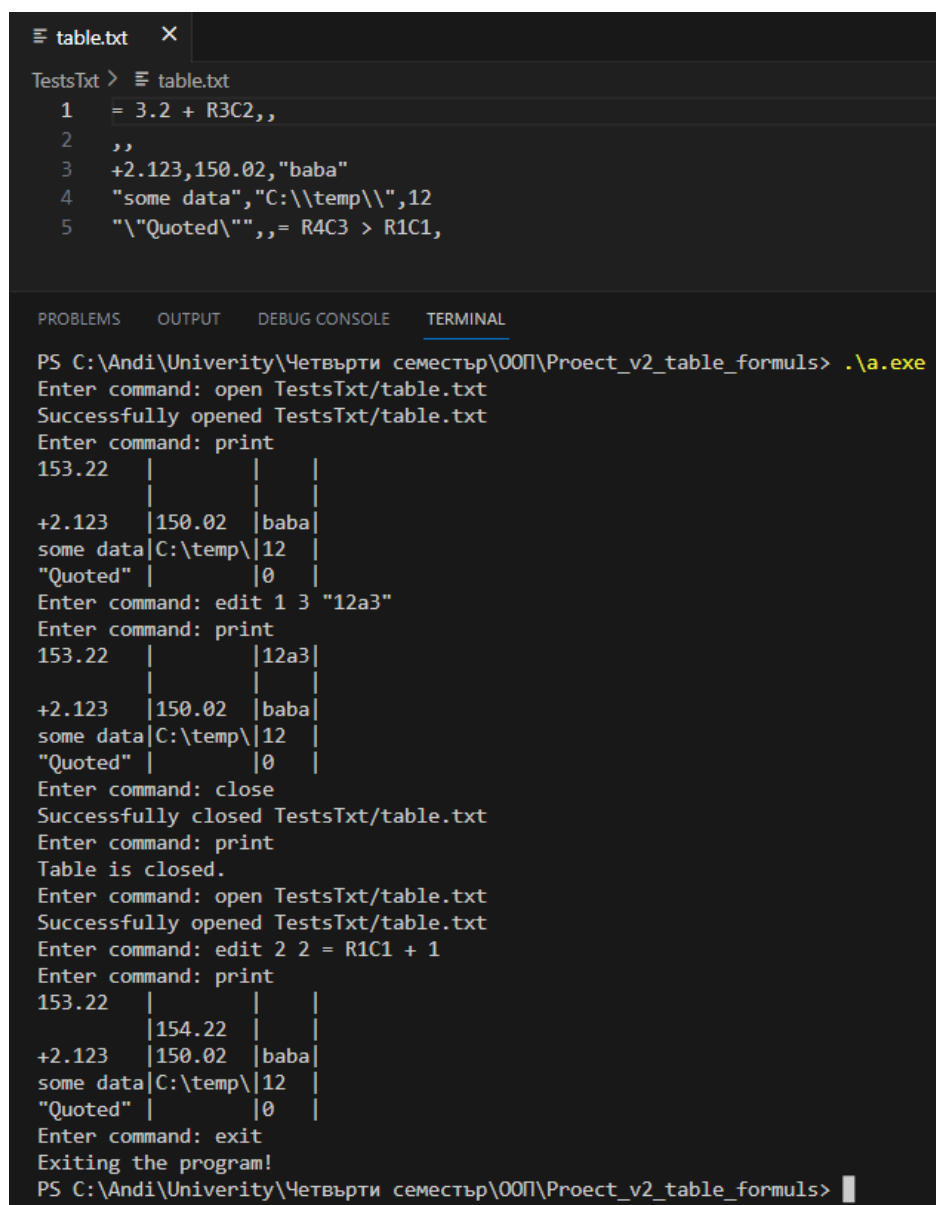
Наследниците на класа Клетка са: Целочислена Клетка, Клетка за дробни числа, Клетка за стринг и Клетка за формули. Всеки от наследниците има параметър за стойност, който е от съответния тип за клетката.

Класът за работа с Таблици е основният клас за програмата. Той съдържа данни от тип вътрешен клас - таблица, в който се съхраняват указатени към клетките и исканите методи за работа с таблицата.

4.2. Управление на паметта и алгоритми. Оптимизации

Основното управление на паметта се извършва от класовете вектор и стринг. Едната явно заделена динамична памет е във функцията за създаване на клетка по подадени стойност и позиция. Функцията се извиква в частен метод на класа Таблица. За освобождаване на тази памет при грешка или при изтриване на обект от класа се грижи друг частен метод.

4.3. Демонстрация



```

table.txt X
TestsTxt > table.txt
1 = 3.2 + R3C2,,
2 ,,
3 +2.123,150.02,"baba"
4 "some data","C:\\temp\\",12
5 "\\Quoted\\",,,= R4C3 > R1C1,

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS C:\\Andi\\Univerity\\Четвърти семестър\\00П\\Проект_v2_table_formuls> .\\a.exe
Enter command: open TestsTxt/table.txt
Successfully opened TestsTxt/table.txt
Enter command: print
153.22 | | |
+2.123 |150.02 |baba|
some data|C:\\temp\\|12 |
"Quoted" | |0 |
Enter command: edit 1 3 "12a3"
Enter command: print
153.22 | |12a3|
+2.123 |150.02 |baba|
some data|C:\\temp\\|12 |
"Quoted" | |0 |
Enter command: close
Successfully closed TestsTxt/table.txt
Enter command: print
Table is closed.
Enter command: open TestsTxt/table.txt
Successfully opened TestsTxt/table.txt
Enter command: edit 2 2 = R1C1 + 1
Enter command: print
153.22 | | |
|154.22 | | |
+2.123 |150.02 |baba|
some data|C:\\temp\\|12 |
"Quoted" | |0 |
Enter command: exit
Exiting the program!
PS C:\\Andi\\Univerity\\Четвърти семестър\\00П\\Проект_v2_table_formuls>

```

Фиг. 4.1 Демонстрацията на работата на програмата

На фиг. 4.1. са демонстриране някои от основните функции за работа с таблици за приложението и примерният текстови файл. Тези функции за open, print, edit и close. Също така може да се види как се принтират формулите.

ПЕТА ГЛАВА

Заклучение

5.1. Обобщение на изпълнението на началните цели

Всички поставени задачи са изпълнени.

5.2. Насоки за бъдещо развитие и усъвършенстване

Идея за бъдещо усъвършенстване на проекта е формулите да поддържат повече от 2 аргумента

ИЗПОЛЗВАНА ЛИТЕРАТУРА

- <https://cplusplus.com/reference/vector/vector/> - документация за vector
- <https://cplusplus.com/reference/string/string/?kw=string> - документация за стринг
- <https://www.geeksforgeeks.org/passing-a-function-as-a-parameter-in-cpp/> - подаване на функция като параметър на друга функция
- <https://cplusplus.com/reference/string/string/?kw=string> - exit() функция
- <https://mathbits.com/MathBits/CompSci/Introduction/clear.htm> - изчистване на конзолата
- <https://learn.fmi.uni-sofia.bg/mod/resource/view.php?id=241316> - работа с текстови файлове