

# Курсов проект по Структури от данни

Специалност „Информационни системи“, 2 курс

Зимен семестър 2012/2013

## Приложение за работа с електронни таблици

### Представяне на данните

Данните на една таблица ще записваме в текстов файл по следния начин:

1. Всеки ред във файла представя отделен ред в таблицата.
2. Всеки ред във файла съдържа данни разделени със запетаи. Тези данни се интерпретират като стойностите в клетките на реда.
3. Всеки ред в таблицата може да съдържа различен брой клетки. Затова и всеки ред във файла може да съдържа различен брой елементи разделени със запетаи.
4. Празен ред във файла представя празен ред в таблицата. (т.е. ред, в който всички клетки са празни)
5. Между две запетаи във файла може да няма никакви данни. По този начин се представя празна клетка.
6. Между данните и запетаите може да има произволен брой празни символи (whitespace).

Така за една таблица може да има различни представяния. Например таблицата:

10	20	30	40
10		1000	
	10		

може да се представи като:

```
10, 20, 30, 40
10,,1000,
,,
,10
```

или

```
10, 20 , 30 , 40
10, , 1000,
, , ,
, 10
```

а също и по други начини.

## Типове данни в таблицата

Клетките в таблицата имат различни типове. Вашето приложение трябва да може да поддържа следните типове:

1. **Цяло число** – поредица от цифри, без никакви други символи между тях. В началото на числото може да има знак '+' или '-'. Например:

```
123  
-123  
+123
```

2. **Реално число** – поредица от цифри, следвана от символ за точка и след нея друга поредица от цифри. В началото на числото може да има знак '+' или '-'. Например:

```
123.456  
-123.456  
+123.456
```

3. **Символен низ (стринг)** – поредица от произволни символи оградени в кавички. Подобно на низовете в C++, ако искате да включите символа за кавичка в даден низ, трябва да го представите като '\', а ако искате да включите наклонена черта, трябва да я представите като '\\'. Например:

```
"Hello world!"  
"C:\\temp\\"  
"\"This is a quotation\""
```

4. **Формула** – формулата винаги започва със символ за равенство. В нея могат да участват следните операции: събиране (+), изваждане (-), умножение (\*), деление (/) и степенуване (^). Във формулата могат да участват или числа или препратки към клетки в таблицата. Ако във формулата участва препратка към клетка, на това място в изчислението трябва да се използва стойността съхранена в дадената клетка. Повече информация за формулите е дадена по-долу.

## Нужна функционалност

**Важно:** Данните за една таблица могат да се въведат в текстов файл с Notepad или друг редактор. Няма нужда вашето приложение да може да редактира файловете.

Вашето приложение трябва да може да:

1. Получи път до текстов файл съдържащ информацията за някаква таблица.
2. Да отвори файла и прочете нужната информация.
3. Да изведе на екрана таблицата.
4. След като данните са изведени на екрана, приложението ви трябва да позволява на потребителя да променя стойностите в отделните клетки. **Приложението ви не трябва да променя оригиналния файл.** Промените се внасят само в зареденото в паметта копие на данните.

Пътят към даден файл съдържащ таблица може да се подаде като аргумент при стартирането на вашата програма. Ако такъв път не бъде подаден, приложението ви трябва да попита потребителя за път към файл и да му позволи да го въведе от клавиатурата. Ако

приложението не успее да отвори файла, то извежда съобщение за грешка и приключва своето изпълнение.

След като приложението ви получи път към файл съдържащ таблица, то трябва да го отвори и да зареди цялата информация за таблицата в паметта. За целта трябва да изберете подходяща структура от данни, в която да съхраните информацията (напр. динамичен масив, свързан списък и т.н.).

Ако при зареждането на данните, приложението ви открие грешка, то трябва да изведе подходящо съобщение за грешка и да прекрати своето изпълнение. Съобщението трябва да подсказва на потребителя какво не е наред във входните данни. Например:

- Ако липсва запетая трябва да се изведе на кой ред и след кой символ липсва запетаята.
- Ако съдържанието на дадена клетка е от неизвестен тип, трябва да се изведе на кой ред и коя колона е клетката и какво точно е некоректното съдържание. Например:
  - На ред 2, колона 5, потребителят е въвел 123.123.123. Ние извеждаме съобщение „Error: row 2, col 5, 123.123.123 is unknown data type”.

Заредената таблица трябва да се изведе на екрана. Данните в колоните трябва да се подравнят. Между отделните колони трябва да се поставят символи за отвесна черта (|). Например ако текстовият файл изглежда така:

```
10, "Hello world!", 123.56
"\Quoted text\"
1, 2, 3, 4
```

той трябва да се изведе на екрана по следния начин:

10		Hello world!		123.56			
"Quoted text"							
1				2		3	
						4	

След като приложението ви зареди и покаже таблицата на екрана, то трябва да позволи на потребителя да променя стойностите на отделните клетки. За целта той трябва да укаже коя клетка иска да промени (на кой ред и коя колона), а също и каква стойност да запише в нея. Потребителят може да въведе произволен тип данни (цяло число, реално число, стринг или формула). Ако той въведе неправилни данни, приложението ви не трябва да променя нищо в таблицата, а само да изведе на екрана съобщение, че са въведени неправилни данни. В този случай приложението ви не трябва да прекратява своето изпълнение. Независимо дали въвеждането на данни в клетката е било успешно или не, след всеки опит за въвеждане, приложението ви трябва отново да изчертае таблицата на екрана.

## Формули

Номерата на редовете и клетките в таблицата започват от 1. Препратка към ред <N> и колона <M> в таблицата се записва така: R<N>C<M>. Например клетката в ред 10 и колона 5 се представя като R10C5.

В дадена формула могат да участват единствено:

1. Литерали: цели или реални числа.
2. Препратки към произволни типове клетки.

При сметките важат следните правила:

1. Ако в дадена формула участват само числа, то сметката се извършва по традиционните правила на аритметиката. Като специален случай можем да отделим делението на две цели числа. В такъв случай не бива да губите остатъка и резултатът трябва да бъде реално число (например 1 делено на 2 дава резултат 0,5).
2. Ако в дадена формула участва низ, той трябва да се конвертира до число. Това става по следния начин: Ако низът съдържа само цифри или поредица от цифри, символ точка и друга поредица от цифри, той се конвертира до съответното число. Всички други низове се конвертират до нула. Например:  
 "123" → 123  
 "123.456" → 123.456  
 "Hello world" → 0  
 "123abc" → 0  
 "123.456.789" → 0
3. Ако в дадена формула участва празна клетка, тя се конвертира до нула. Това важи и за клетки, чиито координати надхвърлят размерите на таблицата.
4. Ако в дадена формула има грешка (например деление на нула), приложението ви не трябва да прекъсва своето изпълнение. Вместо това, когато то извежда таблицата на екрана, в съответната клетка се извежда ERROR, вместо получен резултат.

Примерна таблица (клетките в жълто са от тип число. Клетките в зелено са от тип string):

	Колона 1	Колона 2	Колона 3
Ред 1	10	Hello world!	123.56
Ред 2	123		

Примерни формули и техните оценки:

Формула в клетката	Реално извършена сметка	Стойност на клетката	Коментар
= 10 + 10	10 + 10	20	
= R1C1 + R1C3	10 + 123.56	133.56	
= R1C1 * R1C2	10 * 0	0	Низът „Hello world!“ се конвертира до нула
= R1C1 * R2C1	10 * 123	1230	Низът „123“ се конвертира до 123
= R1C1 * R2C2	10 * 0	0	Клетката на ред 2, колона 2 е празна
= R1C1 * R200C1	10 * 0	0	В таблицата няма ред 200 и колона 200
= 10 / 0	10 / 0	ERROR	
= 10 / R1C2	10 / 0	ERROR	
= R1C1 / R1C2	10 / 0	ERROR	