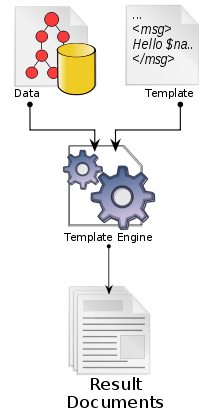
**СУ „Св. Климент Охридски“, ФМИ**

Специалност „Софтуерно Инженерство“

**Увод в програмирането (практикум)**2021-2022 г.

**Задание за курсов проект № 4 - Template engine**



Template engine (процесор за шаблони) се наричат програмите, които използват шаблон и данни в структуриран вид, за да създават документи.

Съставете собствен template engine, който получава на стандартния си вход името на текстовия файл с шаблона и след това името на текстовия файл с данните. Според данните и шаблона запишете получените документи в нова папка results.

На първия ред във файла с данните се подава формата им:

| variable\_name:type,variable\_name\_1:type,... |
| --- |

Където variable\_name може да бъде:

* низ съдържащ малки и големи букви,
* долна черта.

А type показва типа на променливата. Ще се ограничим само до три типа променливи:

* низ,
* масив от низове.

Един примерен формат би имал следния вид:

| id:string,username:string,friends:string\*,messages:string\* |
| --- |

Със string означаваме, че променливата id ще бъде низ, а със string\* - масив от низове.

На всеки следващ ред се намират фактическите данните, с които трябва да се попълва шаблона, например:

| id:string,username:string,friends:string\*,messages:string\*\*  123,teodor,{lili,atanas,teodor},{hi, hello, eho, Hallo}  2,atanas,{lili,atanas,teodor},{hi, hello, eho, Hallo}  17,toni,{lili,atanas,teodor},{hi, hello, eho, Hallo}  ... |
| --- |

Със запетая се отделят стойностите на данните, като скобите {} показват, че имаме масив от стойности.

След обработване на втория ред от примерните данни отгоре трябва да имаме следните променливи:

* id със стойност “1” от тип низ,
* username със стойност “teodor” от тип низ,
* friends със стойност {“lili”, “atanas”, “teodor”} от тип масив от низове,
* messages със стойност {“hi”, “hello”, “eho”, “Hallo”} от тип масив от низове,

които ще се попълват в шаблона.

Шаблонът трябва да може да извежда стойност на променлива и да поддържа оператор за цикъл.

Секциите в шаблона, които ще трябва да се обработват ще се означават с {{ }} и

{% %} .

Извеждането на стойността на дадена променлива в шаблона ще се означава със следния синтаксис: {{ variable\_name }} , което в окончателния документ трябва да се замени със стойността на променливата.

Операторът за цикъл да работи на принципа на [range-based for цикъл](https://www.programiz.com/cpp-programming/ranged-for-loop) и да има следния синтаксис:

| {% for new\_variable in variable\_array %}  тяло на оператора за цикъл  {% endfor %} |
| --- |

Където new\_variable е име на нова променлива, която да бъде с обхват (scope) само в тялото на цикъла, където да може да се използва. variable\_array е променлива от тип масив, чиито елементи ще се итерират. Като по време на итерациите new\_variable приема стойностите на елементите от масива. И за всяка една итерация тялото на цикъла се изпълнява по веднъж.

Важно: При неправилен формат на данните или синтактична грешка в шаблона, програмата трябва да приключи с exit status 1, без да създава нова папка или крайни документи.

Допълнителен пример:

template.txt

| Hello {{ username }},  you have {{ unread\_messages }} new messages:  {% for message in messages %}  - “{{ message }}”  {% endfor %} |
| --- |

database.txt

| username:string,unread\_messages:string,messages:string\*  teodor,2,{Hey Atanas,Hello Georgi,I am waiting Dimitar}  atanas,7,{Hey there! Teodor,Hello Georgi,I am Dimitar,Hi Daniel} |
| --- |

Създайте нова папка results и в нея запишете получените документи. Всеки един от тях трябва да бъде записан в отделен файл, като имената им трябва да спазват следния формат: i.txt , където i е номера на реда във файла с данните, от който са получени.

results/

1.txt:

| Hello teodor,  you have 2 new messages:  - “Hey Atanas”  - “Hello Georgi”  - “I am waiting Dimitar” |
| --- |

2.txt:

| Hello atans,  you have 7 new messages:  - “Hey there! Teodor”  - “Hello Georgi”  - “I am Dimitar”  - “Hi Daniel” |
| --- |