

ЗАДАЧИ ЗА ЗАДЪЛЖИТЕЛНА САМОПОДГОТОВКА

ПО

Структури от данни и програмиране

email: kalin@fmi.uni-sofia.bg

1 декември 2016 г.

1. Да се дефинира `operator *` на шаблона на хеш-таблицата. Хеш-таблицата `s`, която си получава при `s = a * b`, да съдържа като ключове сечението на ключове на `a` и `b`, със съответните им стойности от `b`. Хеш-функцията на `s` да е същата като на `b`.
2. Да се дефинира `operator +` на шаблона на хеш-таблицата. Хеш-таблицата `s`, която си получава при `s = a + b`, да съдържа като ключове симетричната разлика на ключовете на `a` и `b`, със съответните им стойности от `a` и `b`. Хеш-функцията на `s` да е същата като на `b`.

Симетрична разлика на множествата A и B наричаме множеството $C = A \Delta B = A \cup B - A \cap B$, съдържащо тези елементи на A , които не са елементи на B и тези елементи на B , които не са елементи на A .

3. Да се дефинира метод

```
void map (void (*f) (ValueType&))
```

на хеш-таблицата, който прилага функцията `f` над всички стойности в хеш-таблицата.

4. Да се дефинира метод

```
void mapKeys (KeyType (*f) (const KeyType&))
```

на хеш-таблицата, който замества всеки ключ `key` на хеш-таблицата с `f(key)`, като се запазва старата му стойност.

Упътване: Да се извърши съответното ре-хеширане на елемента и той да се премести на съответния нов индекс в таблицата.

5. Методът **begin** на хеш-таблицата да се допълни така, че да може да получава и предикат $p : KeyType \rightarrow bool$. Резултатният итератор да итерираща само през тези ключове от таблицата, които удовлетворяват p .