

*Първо контролно по СДП за специалност Информатика
(11 ноември 2018 г.)*

В тази задача трябва да реализирате основата на специален уеб браузър. Градивните единици на този браузър са "прозорец" и "раздел". За тях в сила са следните правила:

- Броят на "прозорците" не надвишава 255.
- По време на изпълнението на програмата има единствен текущ "прозорец".
- При създаване на нов "прозорец", той скрива предишния и става текущ.
- При затваряне на "прозорец", предходният става текущ (ако има такъв).
- Всеки "прозорец" е съставен от динамичен брой "раздели".
- Няма ограничение за броя на разделите.
- Всеки раздел има времево клеймо, което указва кога е бил създаден. За да вземете текущото време можете да извикате функцията `time()` дефинирана в хедъра `"ctime"` (**).
- Всеки "раздел" изглежда по следния начин:

```
struct Tab {  
    std::string url;           // адрес на отворената в раздела страница  
    time_t timestamp;         // времево клеймо - кога е създаден разделът  
    Tab* pNext;               // указател към следващия раздел  
}
```

** Ако сте забравили как се ползва `<ctime>`, можете да внедрите ваша собствена логика с времената. Например генерирайте поредни цели числа. За това точки **НЯМА** да ви бъдат отнети.

Когато програмата ви се стартира, тя трябва да инициализира браузър с един начален "прозорец", съдържащ единствен "раздел" с URL – `"http://exam_dsa"`. След това тя трябва да може да изпълнява следните команди:

1. **WINDOW OPEN** – отваря нов "прозорец", който се добавя точно след текущия. В него се добавя единствен "раздел" с URL – `"http://exam_dsa"`. Накрая този нов прозорец трябва да стане текущ. Времовата сложност на командата е $O(1)$.

2. **WINDOW CLOSE** – затваря текущия прозорец, като изчиства и всички "раздели" отворени в него. Предходният му "прозорец" става текущ. Ако такъв няма, следващият го става текущ. Ако няма и следващ (това е бил последният прозорец), браузърът остава без прозорци и няма текущ прозорец. Сложността на операцията е $O(N)$ или по-добра, където N е броят на "разделите" в текущия прозорец.

3. UPTIME – показва на екрана колко време е прекарано в текущия "прозорец". Това време се получава като разлика между текущото време и времето клеймо на най-стария му раздел. Сложността на тази команда е $O(1)$.

4. TAB <url> – проверява дали <url> се съдържа в някои от "разделите" на текущия "прозорец". Ако той се съдържа, всички "раздели", започвайки от последното му срещане, се премахват от "прозореца" и командата извежда текст "Trim". Ако <url> не се съдържа, да се добави след последния "таб" и програмата да извежда текст "Append". Накрая командата извежда на екрана всички "раздели" в текущия "прозорец", форматирани по ваша преценка. Сложността на операцията е $O(N)$, където N е броят на "разделите" в текущия прозорец.

5. EXIT – приключва програмата. Преди изхода се извежда статистика за използването на браузъра: брой на всички "раздели", които са отворени в момента и цялото време прекарано в браузъра.

Пример за работата на програмата:

Browser started.

> TAB www.example.com

Append

```
{ http://exam_dsa", 1541416638 }, { "www.example.com",  
1541416945 }
```

> UPTIME

1307 (ако приемем, че UPTIME се извиква точно 1 секунда след добавянето на "www.example.com")

> TAB "abv.bg"

Append

...

> TAB "dir.bg"

Append

...

> TAB "abv.bg"

Trim

```
{“http://exam_dsa”, 1541416638 }, { “www.example.com”,  
1541416945 }, { “abv.bg”, 1541417112 }
```

```
> WINDOW -O
```

New window opened!

```
> TAB “abv.bg”
```

Append

```
{“http://exam_dsa”, 1541416638 }, { “abv.bg”, 1541423241 }
```

```
> WINDOW -O
```

New window opened!

```
> WINDOW -C
```

Current window closed!

```
> TAB www.example.com
```

Append

```
{“http://exam_dsa”, 1541416638 }, { “abv.bg”, 1541423241 }, {  
“www.example.com”, 1541427325 }
```

```
> EXIT
```

Total closed “tabs” : 6. Total time of usage : 6603 (ако EXIT
е изпълнен моментално след последния append)