




Синхронизация на нишки




synchronization. we has it.

ICANHASCHEEZBURGER.COM 🍷 🍷



Проблеми, свързани с нужда от синхронизация

- Понякога трябва да използваме нишки, но те трябва да се изчакват (синхронизират)
- Понякога различни нишки боравят с общ ресурс
- Понякога една нишка е зависима от друга нишка



Класически пример : Много потребители имат достъп до една и съща банкова сметка


наличност 500

потребител 1 – 200;

потребител 2 – 300

Потребителите достъпват сметката едновременно и теглят 200 и 300 единици от съответната валута резултатът е:

$500 - 200 = 300$? $500 - 300 = 200$?



Класически проблем – producer- consumer

Производител произвежда дадена стока, с която зарежда склад с фиксиран капацитет. Клиент пазарува дадената стока в същия момент. Производителят не може да доставя стока, ако складът е пълен, консуматорът не може да пазарува стока, ако складът е празен.

Проблемът придобива още по – голяма сложност при много производители с много потребители.



Използване на lock

- lock създава ключалка, която заключва даден обект, докато трае изпълнението на определен код. Когато изпълнението приключи, обектът се отключва.
- По този начин можем да заключим обекта от текущата нишка и когато тя се изпълни да го освободим
- Другите нишки не могат да достъпят обекта през това време

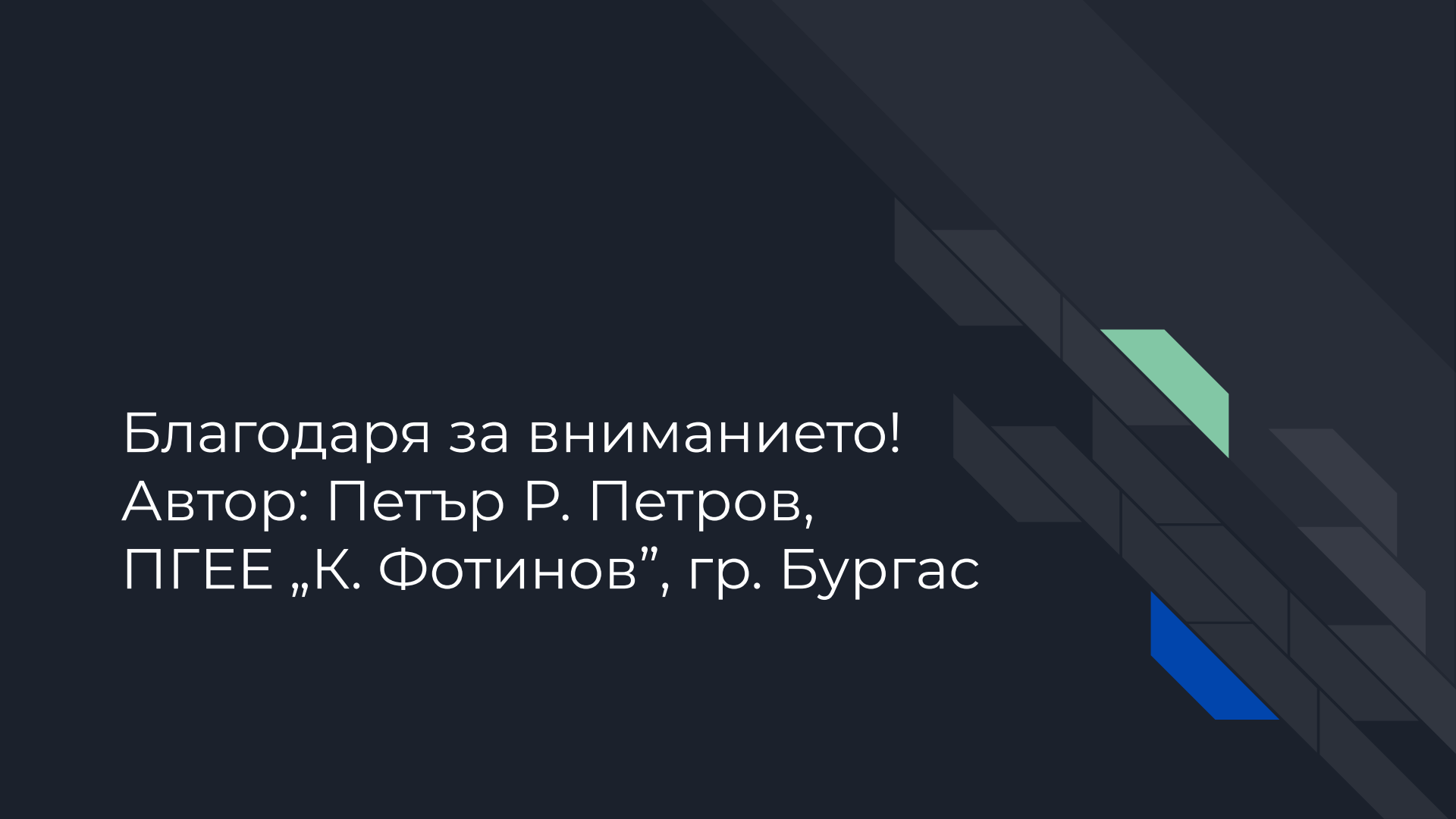
Повече информация за lock:

<https://docs.microsoft.com/en-us/dotnet/csharp/language-reference/keywords/lock-statement>

<https://www.javatpoint.com/c-sharp-thread-synchronization>

<https://www.pluralsight.com/guides/lock-statement-access-data>

Благодаря за вниманието!
Автор: Петър Р. Петров,
ПГЕЕ „К. Фотинов“, гр. Бургас

The background features a series of dark gray, three-dimensional rectangular blocks arranged in a perspective that recedes towards the top right. A light green parallelogram is positioned on one of the upper blocks, and a blue parallelogram is on a lower block further to the right.