

# Стекове



**Учителски екип**

Обучение за ИТ кариера

<https://it-kariera.mon.bg/e-learning/>



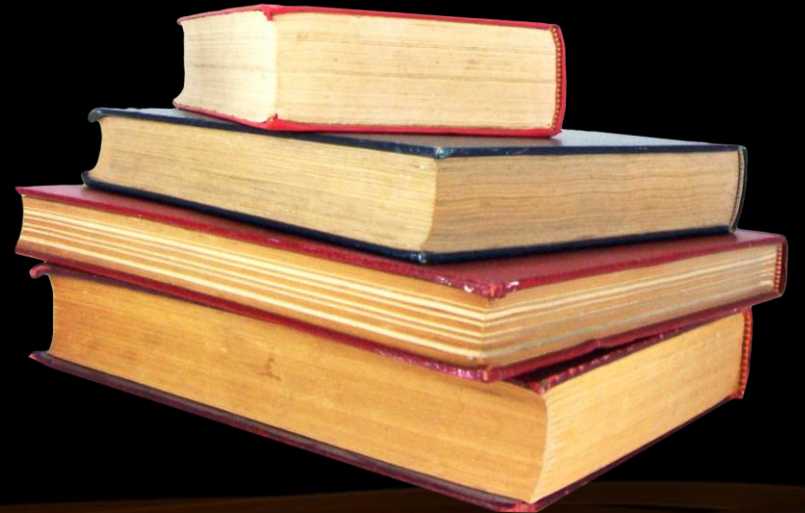
# Съдържание

1. Какво е стек?
2. Статичен стек
3. Динамичен стек
4. Задачи със стекове



# Какво е стек?

- **Стекът** е структура от данни, която има поведение от тип LIFO (Last In, First Out) – „последен влиза, първи излиза“.
  - Т.е. можем да добавяме и извличаме елемент само от “най-горния” край
- **Пример:** колода карти, купчинка чинии или книги в кашон, пакетче с бисквити, програмен стек
- Стекът може да се реализира:
  - **Статично**, чрез масив
  - **Динамично**, чрез възел със стойност и указател към следващ елемент



# Програмен стек

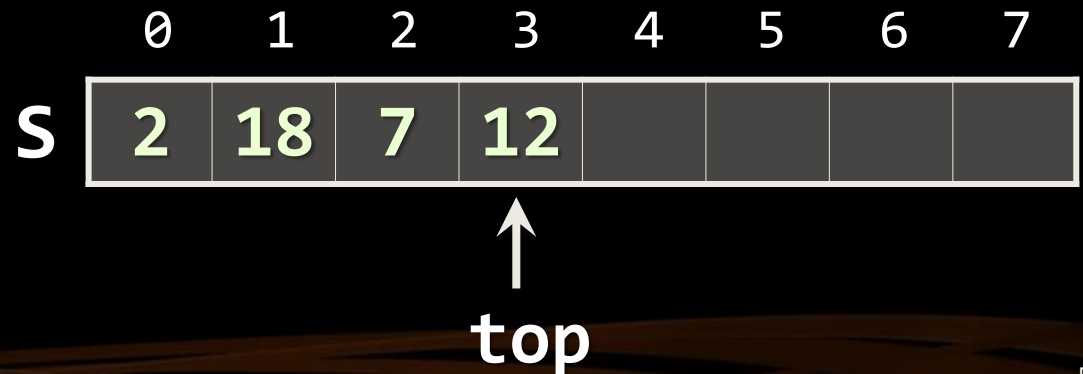
- Стекът е малко парче памет с фиксиран размер (напр. 1MB)
- Пази **локалните променливи** и **точката**, в която всяка активна подпрограма трябва да **върне контрола**, когато **завърши изпълнението си**





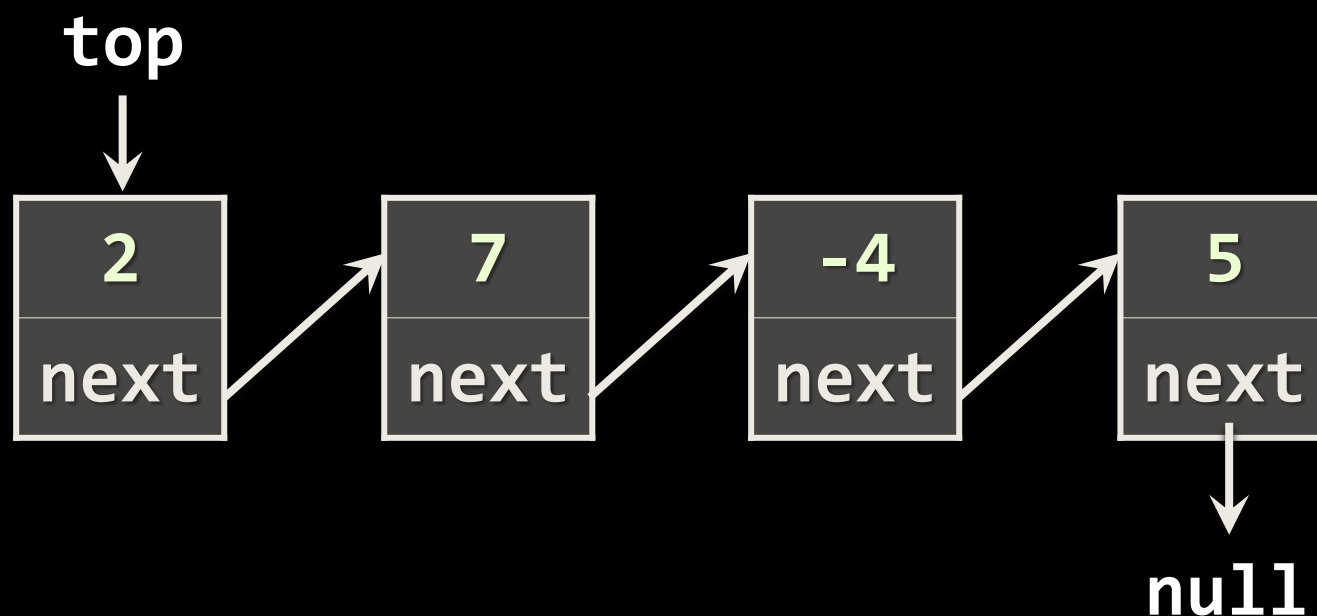
# Статичен стек

- Статична (базирана на масив) имплементация
  - Има фиксиран капацитет
  - Има индекс, който оказва най-горния елемент (**top**), който се движи наляво/надясно според това дали е премахнат / добавен елемент
  - При запълване на капацитета, се заделя двойно място, по принципа на разтегливия масив



# Свързан стек

- Динамична (свързана) реализация
  - Всеки възел (**node**) има 2 полета: **value** и **next**
  - Специален указател съдържа най-горния елемент



# Stack<T> Class в .NET Framework

- Реализиран посредством масив
  - Елементите са от един и същ тип **T**
  - **T** може да бъде всякакъв тип, например: **int** / **Stack<int>** / **Stack<Customer>**
  - Размерът се увеличава автоматично при нужда

# Stack<T>: базова функционалност

- **Push(T)** – добавя елемент към стека

```
stack.Push(5);
```

- **Pop()** – премахва и връща елемента, който е най-горе в стека

```
int number = stack.Pop();
```

- **Peek()** – връща елемента най-горе в стека без да го маха

```
int number = stack.Peek();
```

- **Count** – връща броя елементи в стека

```
int elementCount = stack.Count;
```



## Stack<T>: базова функционалност (2)

- **Clear()** – премахва всички елементи

```
stack.Clear();
```

- **Contains(T)** – проверява дали елемент се среща в стека

```
bool isFound = stack.Contains(5);
```

- **ToArray()** – преобразува стека в обикновен масив

```
int[] arr = stack.ToArray();
```

- **TrimExcess()** – изтрива допълнителното място

```
stack.TrimExcess();
```

# Задача: „Undo“ списък от адреси

- Изведете "посетените отново" страници на браузъра - т.е. заредените след ползване на командата за връщане назад.
- Ще получите възможни команди:
  - URL – отваря дадената страница
  - back – връща към предната страница
  - exit – край на командите

```
www.softuni.bg  
www.judge.softuni.bg  
www.kids.softuni.bg  
back  
back  
exit
```



```
www.judge.softuni.bg  
www.softuni.bg
```

## Решение: „Undo“ списък от адреси

```
if (command == "back")
{
    if (stack.Count != 0) {
        Console.WriteLine(stack.Pop());
    }
    previous = null;
}
else
{
    if (previous != null) {
        stack.Push(previous);
    }
    previous = command;
}
```

# Задача: Съответстващи си квадратни скоби

- Даден е аритметичен израз със скоби (може и вложени)
- Цел: извличане на всички **подизрази** в скоби

$$1 + (2 - (2 + 3) * 4 / (3 + 1)) * 5$$

$$(2 + 3)$$
$$(3 + 1)$$
$$(2 - (2 + 3) * 4 / (3 + 1))$$



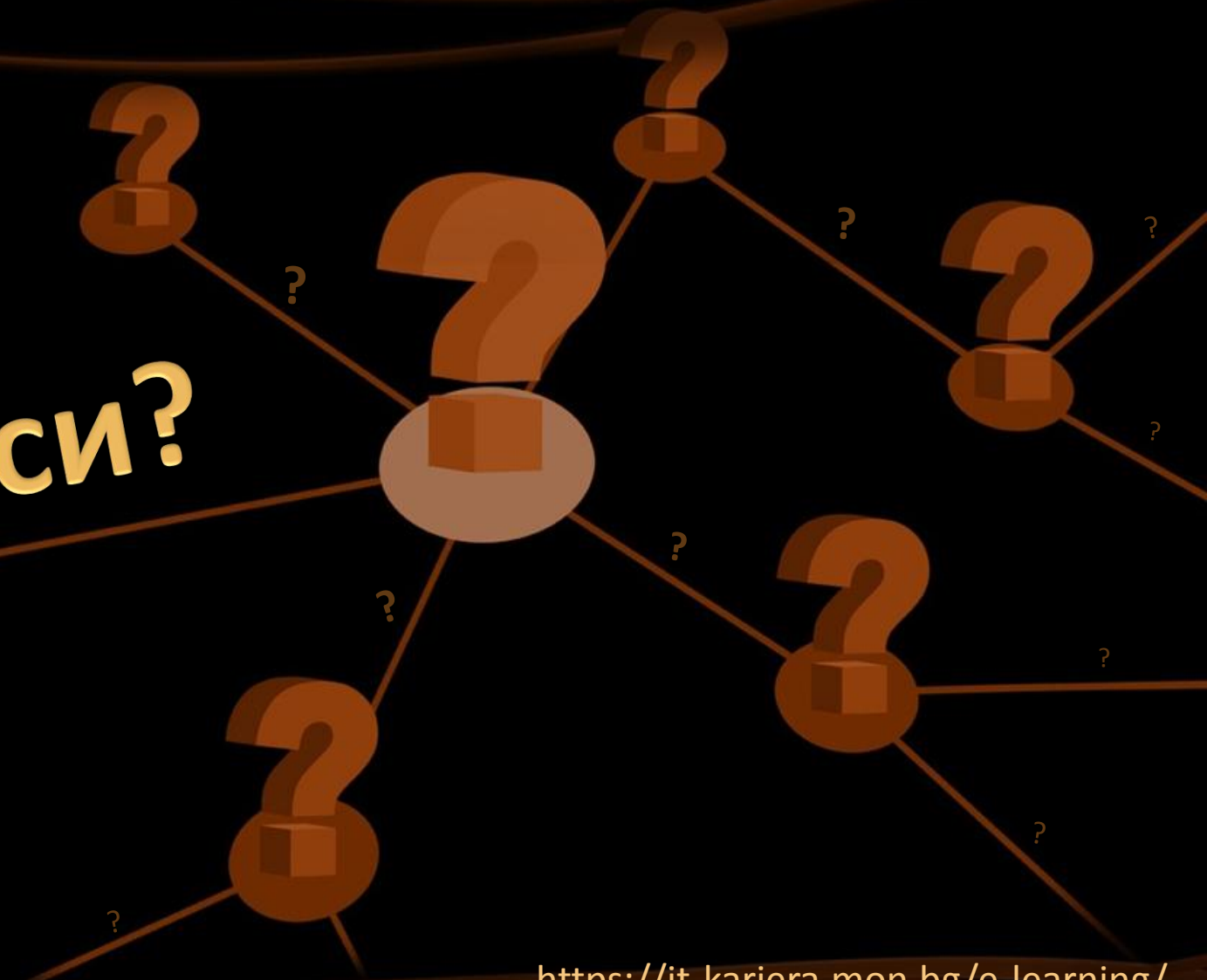
# Решение: Съответстващи си квадратни скоби

```
for (int index = 0; index < expression.Length; index++)
{
    char ch = expression[index];
    if (ch == '(')
        stack.Push(index);
    else if (ch == ')')
    {
        int startIndex = stack.Pop();
        int length = index - startIndex + 1;
        string contents = expression.Substring(startIndex, length);
        Console.WriteLine(contents);
    }
}
```

Стекове



Въпроси?



# Министерство на образованието и науката (МОН)

- Настоящият курс (презентации, примери, задачи, упражнения и др.) е разработен за нуждите на Национална програма "**Обучение за ИТ кариера**" на МОН за подготовка по професия "Приложен програмист"



Министерство  
на образованието  
и науката



Национална  
програма  
„Обучение за  
ИТ кариера“

- Курсът е базиран на учебно съдържание и методика, предоставени от **фондация "Софтуерен университет"** и се разпространява под свободен лиценз **CC-BY-NC-SA**



SoftUni  
Foundation

