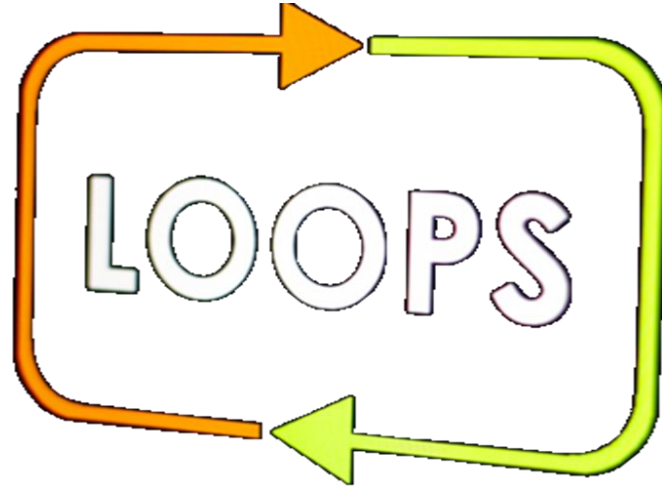


Цикъл for

Прости повторения с For-цикъл

Увод в
програмирането



Съдържание

1. Повторения (цикли)
2. **for**-цикъл в най-простата му форма
 - повторение със стъпка
 - повторение с намаляваща стъпка
3. Задачи с прости повторения
4. Сума на **n** числа, най-голямо и най-малко число



Повторения (цикли)

- В програмирането често пъти се налага да изпълним блок с команди няколко пъти
 - За целта използваме **for**-цикъл

Конструкция
за **for** цикъл

Начална
стойност

Крайна
стойност

Стъпка на
промяна

```
for (var i = 1; i <= 10; i++)
```

```
{
```

```
    Console.WriteLine("i = " + i);
```

```
}
```

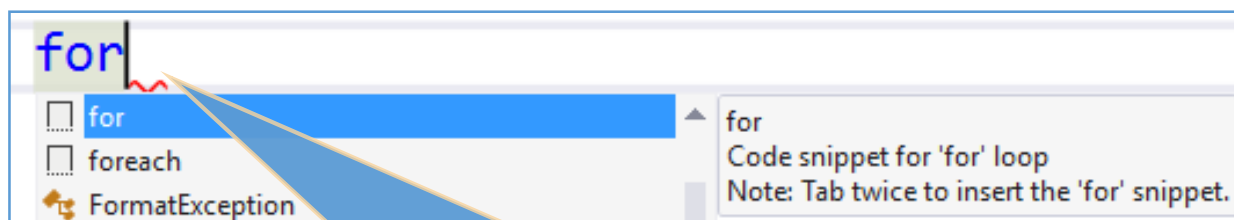
Тяло на цикъла: блок
команди за изпълнение

Пример: числа от 1 до 100

- Да се напише програма, която печата числата от 1 до 100:

```
for (var i = 1; i <= 100; i++)  
{  
    Console.WriteLine(i);  
}
```

- Може да използвате "**for-loop**" **code snippet** във Visual Studio



Натиснете [TAB] два пъти

```
for (int i = 0; i < length; i++)  
{  
}  
}
```

Пример: числа до 1000, завършващи на 7

- Да се напише програма, която намира всички числа в интервала [1...1000], които завършват на 7:

```
for (var i = 1; i <= 1000; i++)  
{  
    if (i % 10 == 7)  
    {  
        Console.Write(i + " ");  
    }  
}
```

Пример: всички латински букви

- Да се напише програма, която отпечатва буквите от латинската азбука: **a** ... **z**
 - **For**-циклите работят не само с числа, може и с букви:

```
Console.Write("Latin alphabet:");  
for (var letter = 'a'; letter <= 'z'; letter++)  
{  
    Console.Write(" " + letter);  
}  
Console.WriteLine();
```

Числата от 1 до N през 3

- Да се отпечатаат числата от **1** до **n** със **стъпка 3**
 - При **n** = 100: **1, 4, 7, 10, ..., 94, 97, 100**

```
var n = int.Parse(Console.ReadLine());  
for (var i = 1; i <= n; i+=3)  
{  
    Console.WriteLine(i);  
}
```



Задаване
на стъпка

Числата от N до 1 в обратен ред

- Да се отпечатаат числата от **n** до **1** в обратен ред (стъпка **-1**)
 - При **n** = 100: **100, 99, 98, ..., 3, 2, 1**

```
var n = int.Parse(Console.ReadLine());  
for (var i = n; i >= 1; i -= 1)  
{  
    Console.WriteLine(i);  
}
```

Обърнато
условие:
i >= 1

Отрицателна
стъпка: **-1**



Числата от 1 до 2^n с for-цикъл

- Да се отпечатаат числата от **1** до **2^n**
 - При **n** = 10: **1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024**

```
var num = 1;
for (var i = 0; i <= n; i++)
{
    Console.WriteLine(num);
    num = num * 2;
}
```

| | | | |
|----|-----|---|---|
| 8 | 2 | 8 | |
| 16 | 128 | 4 | 2 |
| 2 | 4 | 2 | 4 |
| 2 | 32 | 8 | |

Четни степени на 2

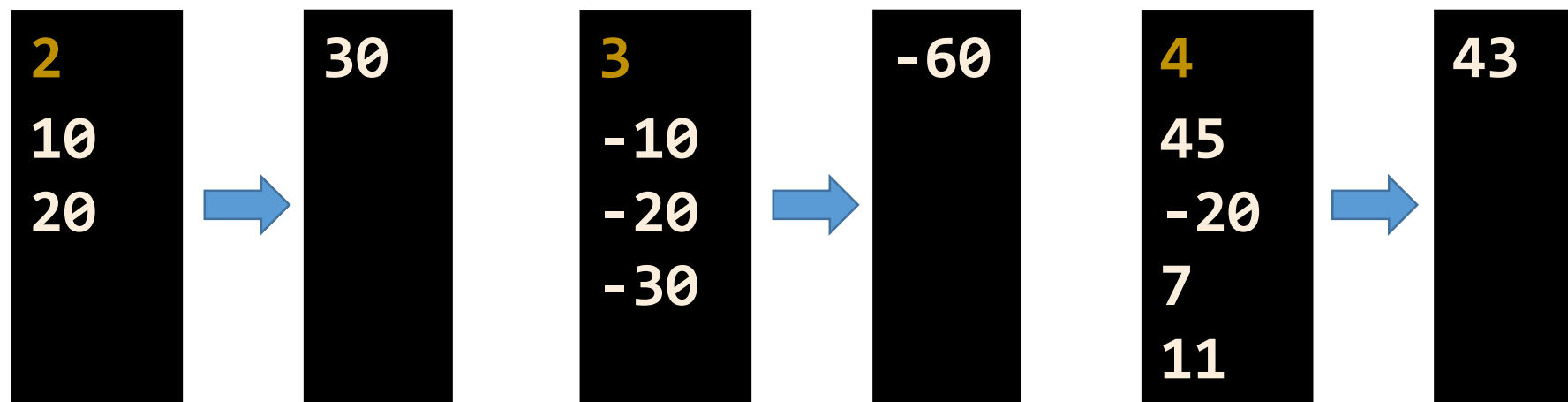
- Да се отпечатаат **четните степени на 2** до 2^n : $2^0, 2^2, 2^4, 2^8, \dots, 2^n$
 - При $n = 10$: **1, 4, 16, 64, 256, 1024**

```
var num = 1;
for (var i = 0; i <= n; i+=2)
{
    Console.WriteLine(num);
    num = num * 2 * 2;
}
```

Ползваме
стъпка **2**

Пример: сумиране на числа

- Да се напише програма, която въвежда **n** числа и ги **сумира**
 - От първия ред на входа се въвежда броят числа **n**
 - От следващите **n** реда се въвежда по едно число
 - Числата се **сумират** и накрая се отпечатва резултатът
 - Примери:



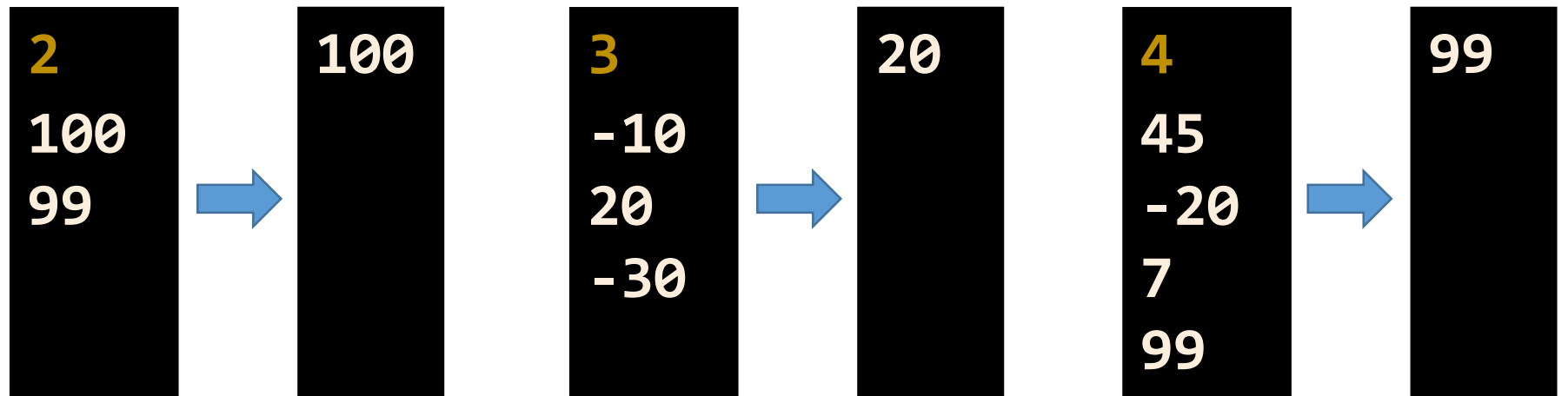
Решение: сумиране на числа

```
Console.Write("n = ");  
var n = int.Parse(Console.ReadLine());  
Console.WriteLine("Enter the numbers:");  
  
var sum = 0;  
for (var i = 0; i < n; i++)  
{  
    var num = int.Parse(Console.ReadLine());  
    sum = sum + num;  
}  
  
Console.WriteLine("sum = " + sum);
```

Пример: най-голямо число

- Да се напише програма, която въвежда **n** числа и намира **най-голямото** измежду тях
 - От първия ред на входа въвежда броя числа **n**
 - От следващите **n** реда се въвежда по едно число

- Примери:

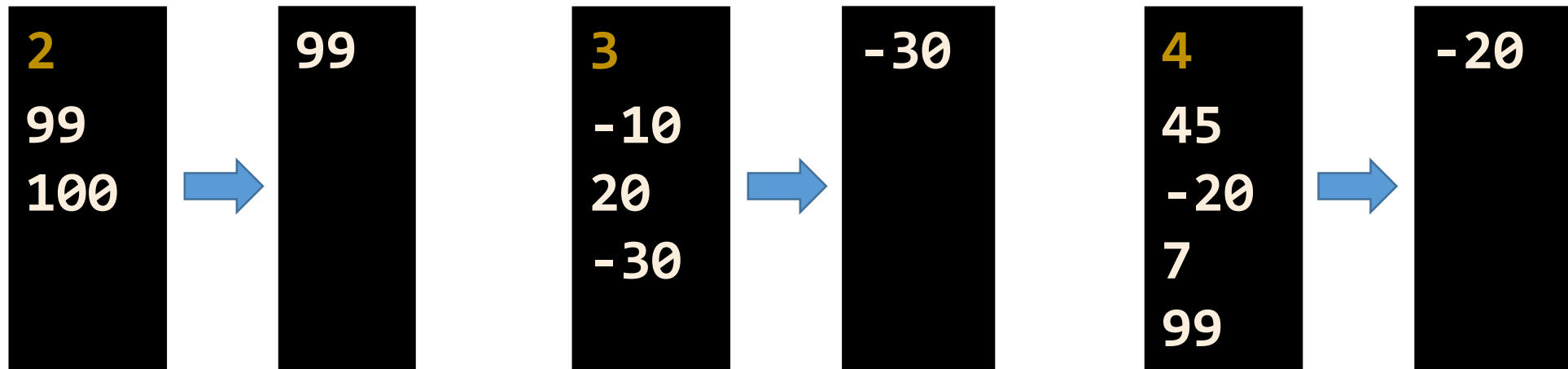


Решение: най-голямо число

```
Console.Write("n = ");
var n = int.Parse(Console.ReadLine());
var max = -1000000000000000;
for (var i = 0; i < n; i++)
{
    var num = int.Parse(Console.ReadLine());
    if (num > max)
        max = num;
}
Console.WriteLine("max = " + max);
```

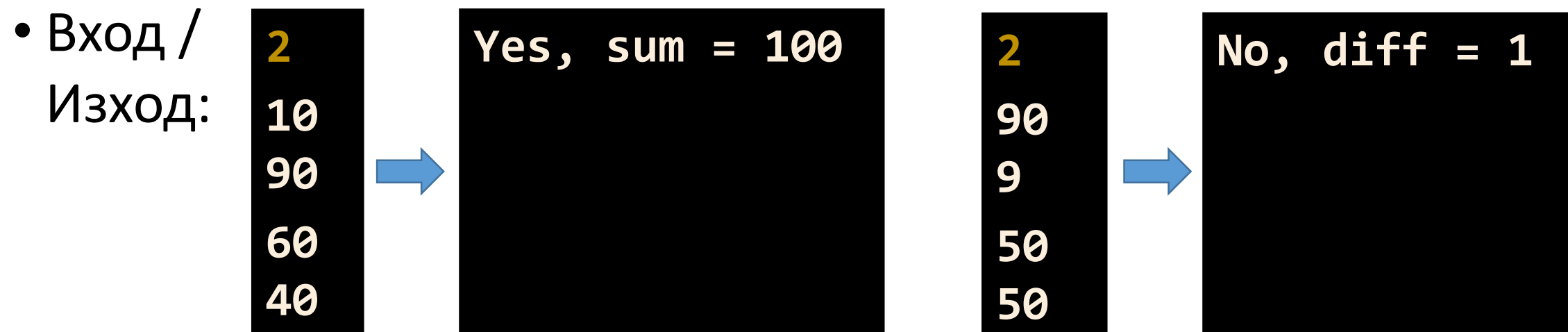
Пример: най-малко число

- Да се напише програма, която въвежда **n** числа и намира **най-малкото** измежду тях
 - Въвежда първо броя числа **n**, след тях още **n** числа
- Примери:



Задача: лява и дясна сума

- Да се напише програма, която въвежда $2*n$ числа
 - Проверява дали сумите на левите n и десните n числа са равни
 - При равенство печата "Yes" + сумата; иначе печата "No" + разликата (изчислена като положително число)

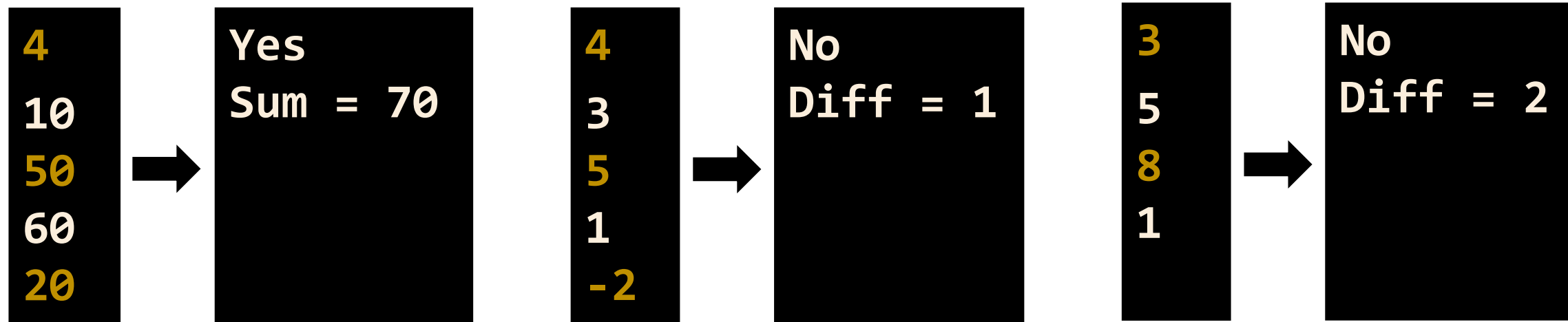


Решение: лява и дясна сума

```
var n = int.Parse(Console.ReadLine());
var leftSum = 0;
for (var i = 0; i < n; i++)
    { leftSum = leftSum + int.Parse(Console.ReadLine()); }
// TODO: read and calculate the rightSum
if (leftSum == rightSum)
    { Console.WriteLine("Yes, sum = " + leftSum); }
else
    { Console.WriteLine("No, diff = " +
        Math.Abs(rightSum - leftSum)); }
```

Задача: четна / нечетна сума

- Да се напише програма, която въвежда **n** числа
 - Проверява дали сумата на числата на **четни позиции** е равна на сумата на числата на **нечетни позиции**
 - При равенство печата "**Yes**" + **сумата**; иначе печата "**No**" + **разликата** (положително число). Примери:



Решение: четна / нечетна сума

```
var n = int.Parse(Console.ReadLine());
var oddSum = 0;
var evenSum = 0;
for (var i = 0; i < n; i++)
{
    var element = int.Parse(Console.ReadLine());
    if (i % 2 == 0) oddSum += element;
    else evenSum += element;
}

// TODO: print the sum / difference
```

Какво научихме днес?

- Можем да повтаряме блок код с **for**-цикъл:

```
for (var i = 1; i <= 10; i++)  
{ Console.WriteLine("i = " + i); }
```

- Можем да четем поредица от **n** числа от конзолата:

```
var n = int.Parse(Console.ReadLine());  
for (var i = 0; i < n; i++)  
{ var num = int.Parse(Console.ReadLine()); ... }
```

- Можем да ползваме **for**-цикли със **стъпка**:

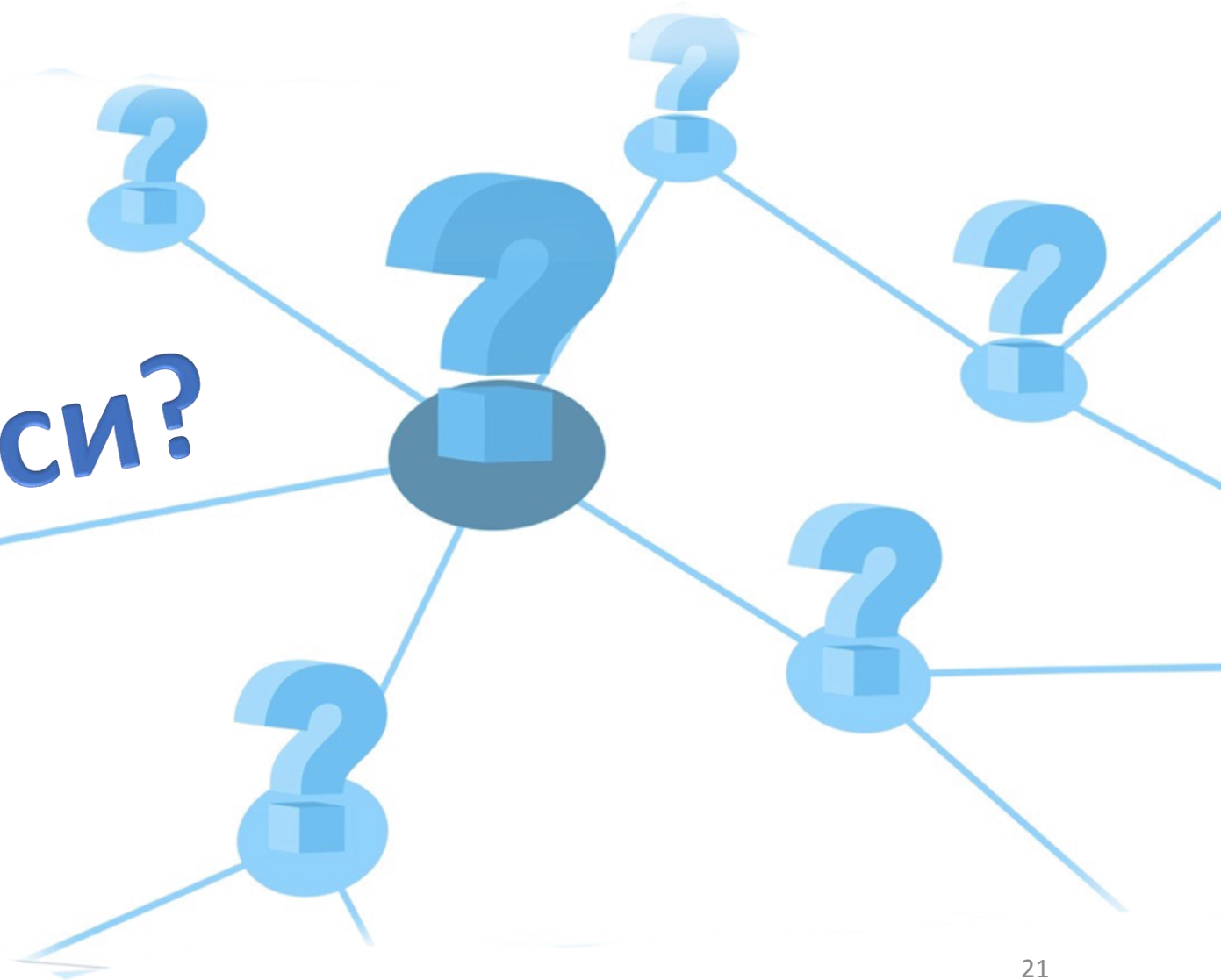
```
for (var i = 1; i <= n; i+=3) Console.WriteLine(i);
```



Цикъл for



Въпроси?



Договор за ползване

Този курс (слайдове, примери, задачи и др.) се разпространяват под свободен лиценз "[Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International](https://creativecommons.org/licenses/by-nc-sa/4.0/)"



Базиран е на учебните материали на [НП „Обучение за ИТ Кариера“](#).

Може да съдържа части от следните източници:

- Книга "[Основи на програмирането със C#](#)" от Светлин Наков и колектив с лиценз [CC-BY-SA](#)