

**ОТГОВОРИ, УПЪТВАНИЯ И ПРИМЕРНИ ИЗГЛЕДИ  
НА РЕШЕНИЯТА НА ПРАКТИЧЕСКИТЕ  
ЗАДАЧИ ОТ**

**ТЕМА 11**

<b>Въпрос</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>	<b>13</b>	<b>14</b>	<b>15</b>	<b>16</b>
<b>Отговор</b>	<b>Г</b>	<b>Г</b>	<b>А</b>	<b>Б</b>	<b>А, Б и В</b>	<b>А</b>	<b>А</b>	<b>Б</b>	<b>В</b>	<b>Б</b>	<b>Г</b>	<b>В</b>	<b>Г</b>	<b>Б</b>	<b>Б</b>	<b>В</b>

**17.**

<b>C#</b>
<pre>using System; class Program {     static void Main()     {         int[] numbers = new int[] { 89, 4, 23, 33, 3, 85, 40, 18, 23, 75, 21, 57, 2, 75, 84, 18, 59, 53, 16, 84 };         Console.WriteLine("Нечетни числа:");         for (int i = 0; i &lt; numbers.Length; i++)         {             if (numbers[i] % 2 != 0)                 Console.Write("{0} ", numbers[i]);         }     } }</pre>
<b>Java</b>
<pre>public class Main {     public static void main(String[] args) {         int[] numbers = { 89, 4, 23, 33, 3, 85, 40, 18, 23, 75, 21, 57, 2, 75, 84, 18, 59, 53, 16, 84 };         System.out.println("Нечетни числа:");         for (int i = 0; i &lt; numbers.length; i++) {             if (numbers[i] % 2 != 0)                 System.out.print(numbers[i] + " ");         }     } }</pre>

**18.**

<b>C#</b>
<pre>using System; class Program {     static void Main()     {         string[] words = { "химикал", "тетрадка", "лист", "калкулатор" };     } }</pre>

```

string longest = "";
int letterCount = 0;

foreach (string item in words)
{
    if (item.Length > letterCount)
    {
        letterCount = item.Length;
        longest = item;
    }
}
Console.WriteLine($"Най-дългата дума: {longest}, Брой на
буквите: {letterCount}");
}
}

```

#### Java

```

public class Main {
    public static void main(String[] args) {
        String[] words = { "химикал", "тетрадка", "лист",
"калкулатор" };
        String longest = "";
        int letterCount = 0;

        for (String item : words) {
            if (item.length() > letterCount) {
                letterCount = item.length();
                longest = item;
            }
        }
        System.out.println("Най-дългата дума: " + longest + ",
Брой на буквите: " + letterCount);
    }
}

```

#### 19.

С#	Java
1x0 = 0	
1x1 = 1	
1x2 = 2	
1x3 = 3	
1x4 = 4	
1x5 = 5	
1x6 = 6	
1x7 = 7	
1x8 = 8	
1x9 = 9	
1x10 = 10	
.....	
Таблицата за умножение до 10	

20. А, Д

21. – 20 10 5

22. В

23.

SELECT name

FROM products

WHERE category = 'Дрехи' AND price BETWEEN 50 AND 100

ORDER BY price ASC;

24.

(1) Колко е броят на операциите на изваждане и добавяне? Отговор: 14

(2) Колко са елементите в стека след изпълнението на операциите? Отговор: 7

(3) Кои са елементите на стека след изпълнението на операциите? Отговор:

-1, 2, 3, 0, -2, 1, -4

25.

• Създайте таблица **Students** и таблица **Courses**, въведете съответните записи.

Създаване на таблицата **Students**

CREATE TABLE IF NOT EXISTS Students (

StudentId INT PRIMARY KEY,

FirstName VARCHAR(50),

LastName VARCHAR(50),

BirthDate DATE,

Major VARCHAR(50),

GPA DECIMAL(3, 2)

);

Вмъкване на записи за студенти

INSERT INTO Students (StudentId, FirstName, LastName, BirthDate, Major,

GPA)

VALUES

(1, 'Иван', 'Петров', '2000-05-15', 'Computer Science', 3.75),

(2, 'Мария', 'Иванова', '1999-03-20', 'Engineering', 3.90),

(3, 'Георги', 'Ковачев', '2001-01-10', 'Mathematics', 3.60);

Създаване на таблицата **Courses**

CREATE TABLE IF NOT EXISTS Courses (

CourseId INT PRIMARY KEY,

CourseName VARCHAR(100),

Instructor VARCHAR(50),

StartDate DATE,

EndDate DATE,

Credits INT

);

Вмъкване на записи за курсове

```
INSERT INTO Courses (CourseId, CourseName, Instructor, StartDate, EndDate, Credits)
```

```
VALUES
```

```
(101, 'Алгоритми и структури от данни', 'Проф. Иванов', '2021-09-01', '2022-01-15', 4),
```

```
(102, 'Инженерна математика', 'Проф. Георгиева', '2021-08-15', '2022-01-31', 3),
```

```
(103, 'Програмиране на Python', 'Доц. Петров', '2021-10-01', '2022-02-28', 3);
```

- Коригирайте средния успех (GPA) на студент с идентификатор 3.90 на 3.80:

```
UPDATE Students
```

```
SET GPA = 3.80
```

```
WHERE StudentId = 2;
```

- Извлечете информация за всички студенти, които учат по специалност 'Computer Science':

```
SELECT *
```

```
FROM Students
```

```
WHERE Major = 'Computer Science';
```

- Брой на курсовете, които започват след 1 януари 2022 година:

```
SELECT COUNT(*)
```

```
FROM Courses
```

```
WHERE StartDate > '2022-01-01';
```

- Списък с имената на курсовете и инструкторите им, сортирани по азбучен ред на имената на курсовете, за тези от тях, които вече са приключили:

```
SELECT CourseName, Instructor
```

```
FROM Courses
```

```
WHERE EndDate < CURRENT_DATE
```

```
ORDER BY CourseName;
```

**C#**

```

using System;
using System.Collections.Generic;
class Book
{
    public string Title { get; set; }
    public string Author { get; set; }
    public int Year { get; set; }
    public Book(string title, string author, int year)
    {
        Title = title;
        Author = author;
        Year = year;
    }
    public override string ToString()
    {
        return $"Заглавие: {Title}, Автор: {Author}, Година:
{Year}";
    }
}
class Library
{
    private List<Book> books;
    public Library()
    {
        books = new List<Book>();
    }
    public void AddBook(Book book)
    {
        books.Add(book);
    }
    public void RemoveBook(string title)
    {
        Book bookToRemove = books.Find(book => book.Title ==
title);
        if (bookToRemove != null)
        {
            books.Remove(bookToRemove);
        }
    }
    public void ListBooks()
    {
        foreach (var book in books)
        {
            Console.WriteLine(book);
        }
    }
}
class Program
{
    static void Main()
    {

```

```

        Console.OutputEncoding = System.Text.Encoding.UTF8;
        Library library = new Library();
        library.AddBook(new Book("Майстор и Маргарита", "Михаил
Булгаков", 1967));
        library.AddBook(new Book("Хобит", "Дж.Р.Р. Толкин",
1937));
        library.AddBook(new Book("Цветя за Алжернон", "Даниел
Кийс", 1959));
        Console.WriteLine("Извеждане на книгите в
библиотеката:");
        library.ListBooks();
        library.RemoveBook("Майстор и Маргарита");
        Console.WriteLine("Извеждане след премахване на книга:");
        library.ListBooks();
    }
}

```

## Java

```

import java.util.ArrayList;
import java.util.List;
class Book {
    private String title;
    private String author;
    private int year;
    public Book(String title, String author, int year) {
        this.title = title;
        this.author = author;
        this.year = year;
    }
    @Override
    public String toString() {
        return "Заглавие: " + title + ", Автор: " + author + ",
Година: " + year;
    }
}
class Library {
    private List<Book> books;
    public Library() {
        books = new ArrayList<>();
    }
    public void addBook(Book book) {
        books.add(book);
    }
    public void removeBook(String title) {
        books.removeIf(book -> book.toString().contains(title));
    }
    public void listBooks() {
        for (Book book : books) {
            System.out.println(book);
        }
    }
}

```

```

    }
}
public class Main {
    public static void main(String[] args) {
        System.out.println("Извеждане на книгите в
библиотеката:");
        Library library = new Library();
        library.addBook(new Book("Майстор и Маргарита", "Михаил
Булгаков", 1967));
        library.addBook(new Book("Хобит", "Дж.Р.Р. Толкин",
1937));
        library.addBook(new Book("Цветя за Алжернон", "Даниел
Кийс", 1959));
        library.listBooks();
        System.out.println("Извеждане след премахване на
книга:");
        library.removeBook("Майстор и Маргарита");
        library.listBooks();
    }
}

```

27.

**C#**

```

using System;
class Program
{
    static void Main()
    {
        try
        {
            Console.Write("Масив: ");
            string input = Console.ReadLine();
            string[] inputArray = input.Split(',', ' ',
StringSplitOptions.RemoveEmptyEntries);
            int[] numbers = new int[inputArray.Length];
            for (int i = 0; i < inputArray.Length; i++)
            {
                if (!int.TryParse(inputArray[i], out numbers[i]))
                {
                    throw new FormatException("Invalid input
format. Please enter valid integers separated by commas or
spaces.");
                }
            }
            FindMinMax(numbers);
        }
        catch (Exception ex)
        {
            Console.WriteLine("Something went wrong! " + ex.Message);
        }
    }
}

```

```

    }
}
static void FindMinMax(int[] array)
{
    if (array.Length == 0)
    {
        throw new InvalidOperationException("The array is empty.");
    }
    int min = array[0];
    int max = array[0];
    for (int i = 1; i < array.Length; i++)
    {
        if (array[i] < min)
        {
            min = array[i];
        }
        if (array[i] > max)
        {
            max = array[i];
        }
    }
    Console.WriteLine("Най-голямото число: " + max);
    Console.WriteLine("Най-малкото число: " + min);
}
}

```

#### Java

```

import java.util.Arrays;
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {
        try {
            Scanner scanner = new Scanner(System.in);
            System.out.print("Масив: ");
            String input = scanner.nextLine();
            String[] inputArray = input.split("[,\\s]+");
            int[] numbers = new int[inputArray.length];
            for (int i = 0; i < inputArray.length; i++) {
                numbers[i] = Integer.parseInt(inputArray[i]);
            }
            findMinMax(numbers);
        } catch (Exception ex) {
            System.out.println("Something went wrong! " +
ex.getMessage());
        }
    }
    static void findMinMax(int[] array) {
        if (array.length == 0) {
            throw new IllegalStateException("The array is
empty.");
        }
    }
}

```



```

        int min = array[0];
        int max = array[0];
        for (int i = 1; i < array.length; i++) {
            if (array[i] < min) {
                min = array[i];
            }
            if (array[i] > max) {
                max = array[i];
            }
        }
        System.out.println("Най-голямото число: " + max);
        System.out.println("Най-малкото число: " + min);
    }
}

```

## 28.

**C#**

```

using System;
using System.Collections.Generic;
public class Product
{
    public string Name { get; }
    public double Price { get; }
    public Product(string name, double price)
    {
        if (string.IsNullOrWhiteSpace(name) || price < 0)
        {
            throw new ArgumentException("Invalid name or price");
        }
        Name = name;
        Price = price;
    }
}
public class ProductList
{
    private List<Product> products;
    public ProductList()
    {
        products = new List<Product>();
    }
    public int Size => products.Count;
    public Product Get(int index)
    {
        if (index < 0 || index >= products.Count)
        {
            throw new IndexOutOfRangeException("Invalid index");
        }
        return products[index];
    }
    public void Add(Product product)
    {
        if (products.Contains(product))
        {

```

```

        throw new ArgumentException("Product already exists
in the list");
    }
    products.Add(product);
    products.Sort((p1, p2) => p1.Name.CompareTo(p2.Name));
}
}
class MainClass
{
    public static void Main(string[] args)
    {
        try
        {
            Console.Write("Enter the number of products: ");
            int n = int.Parse(Console.ReadLine());
            ProductList productList = new ProductList();
            for (int i = 0; i < n; i++)
            {
                Console.Write($"Enter product {i + 1} details
(name price): ");
                string[] productDetails = Console.ReadLine().
Split();
                if (productDetails.Length != 2)
                {
                    throw new ArgumentException("Invalid input
format");
                }
                string name = productDetails[0];
                double price = double.Parse(productDetails[1]);
                productList.Add(new Product(name, price));
            }
            Console.WriteLine("ProductList:");
            for (int i = 0; i < productList.Size; i++)
            {
                Product product = productList.Get(i);
                Console.WriteLine($"{i + 1}. {product.Name}
({product.Price:C2})");
            }
        }
        catch (Exception ex)
        {
            Console.WriteLine($"Error: {ex.Message}");
        }
    }
}

```

## Java

```
import java.util.ArrayList;
import java.util.Collections;
import java.util.Comparator;
import java.util.List;
import java.util.Scanner;
class Product {
    private final String name;
    private final double price;
    public Product(String name, double price) {
        if (name == null || name.trim().isEmpty() || price < 0) {
            throw new IllegalArgumentException("Invalid name or
price");
        }
        this.name = name;
        this.price = price;
    }
    public String getName() {
        return name;
    }
    public double getPrice() {
        return price;
    }
}
class ProductList {
    private final List<Product> products;
    public ProductList() {
        products = new ArrayList<>();
    }
    public int size() {
        return products.size();
    }
    public Product get(int index) {
        if (index < 0 || index >= products.size()) {
            throw new IndexOutOfBoundsException("Invalid index");
        }
        return products.get(index);
    }
    public void add(Product product) {
        if (products.contains(product)) {
            throw new IllegalArgumentException("Product already
exists in the list");
        }
        products.add(product);
        Collections.sort(products, Comparator.
comparing(Product::getName));
    }
}
```

```

}
class Main {
    public static void main(String[] args) {
        try {
            Scanner scanner = new Scanner(System.in);
            System.out.print("Enter the number of products: ");
            int n = Integer.parseInt(scanner.nextLine());
            ProductList productList = new ProductList();
            for (int i = 0; i < n; i++) {
                System.out.print("Enter product " + (i + 1) + "
details (name price): ");
                String name = scanner.next();
                double price = scanner.nextDouble();
                productList.add(new Product(name, price));
            }
            System.out.println("ProductList:");
            for (int i = 0; i < productList.size(); i++) {
                Product product = productList.get(i);
                System.out.println((i + 1) + ". " + product.
getName() + " (" + product.getPrice() + ")");
            }
        } catch (Exception ex) {
            System.out.println("Error: " + ex.getMessage());
        }
    }
}

```