

ОБУЧЕНИЕ ПО ПРОГРАМИРАНЕ В ПОДГОТОВКАТА НА УЧИТЕЛИ

Коста Гъров, Стефка Анева, Елена Тодорова

Резюме. Настоящата работа е посветена на обучението по програмиране при подготовката на учители във Факултета по математика и информатика на Пловдивски университет „Паисий Хилендарски“. Учебните курсове по програмиране заемат централно място в учебните програми за бакалавърските специалности „Математика, информатика и информационни технологии“ и „Информационни технологии, математика и образователен мениджмънт“, магистърските програми „Обучение по информатика и информационни технологии в училище“ и „Педагогика на обучението по информатика и информационни технологии“ и квалификация „Учител по информатика и информационни технологии“. В работата са анализирани актуалните учебни програми по информатика за съответните специалности и магистърски програми. Представени са идеи за методики на преподаване на програмиране и компютърно моделиране, свързани със средното училище. Показани са подходи за създаване на елементи на методически инструментариум. Ключови думи: програмиране, информатика, компютърно моделиране, методика на обучение.

Ключови думи: програмиране, информатика, компютърно моделиране, методика на обучение.

1. Въведение

Факултетът по математика и информатика към ПУ „Паисий Хилендарски“ е факултетът в България, в който се обучават най-много студенти – бъдещи учители по математика, информатика и информационни технологии. В момента се обучават 141 редовни студенти в бакалавърски програми, 184 студенти в магистърски програми и 70 – в допълнителни квалификации.

Основни задължителни учебни дисциплини при обучението на бъдещите учители по информатика и ИТ в бакалавърски специалности „Математика, информатика и информационни технологии“ (МИИТ) и „Информационни технологии, математика и образователен мениджмънт“ (ИТМОМ), свързани с изучаване на програмиране са представени в Таблица 1.

Таблица 1. Учебни курсове, свързани с програмиране в бакалавърски специалности МИИТ и ИТМОМ

Година от обучението	Наименование на дисциплина
Първа година	Програмиране; Обектно-ориентирано програмиране Въведение в уеб програмирането
Втора година	Алгоритми и структури от данни
Трета година	Информационни технологии в образованието Компютърна графика и презентации
Четвърта година	Училищен курс по информатика Училищен курс по информационни технологии Събитийно програмиране Интегриран практикум по информатика и информационни технологии

Основни задължителни учебни дисциплини при обучението на бъдещите учители по информатика и ИТ в магистърски програми, свързани с изучаване на програмиране са представени в Таблица 2.

Таблица 2. Учебни курсове, свързани с програмиране в магистърски програми, специалност „Обучение по информатика и ИТ в училище“ 2 г.

Година от обучението	Наименование на дисциплина
Първа година	Училищен курс по информатика Училищен курс по информационни технологии
Втора година	Проектиране и приложение на алгоритми Блоково програмиране Приложения с графичен интерфейс Уеб технологии Основна специализираща дисциплина „Компетентностен подход и иновации в образованието“

Основни задължителни учебни дисциплини при обучението на бъдещите учители по информатика и ИТ в допълнителна квалификация УИИТ, свързани с изучаване на програмиране са представени в Таблица 3.

Таблица 3. Учебни курсове, свързани с програмиране в допълнителна квалификация „Учител по информатика и ИТ“, 1 г.

Година от обучението	Наименование на дисциплина
Първа година	Програмиране; Интегриран практикум по информатика Училищен курс по информатика и информационни технологии Основна специализираща дисциплина „Компетентностен подход и иновации в образованието“

Анализът на учебните програми, свързани с програмиране, по които се извършва подготовката на бъдещи учители във ФМИ показва, че основните теми на учебно съдържание са: Програмиране; Обектно-ориентирано програмиране; Алгоритми и структури от данни; Събитийно програмиране; Блоково програмиране; Програмиране на скриптов езици; Уеб програмиране. Вижда се, че изучаваните дисциплини са сравнително сложни и изискват използването на допълнителни методики за преподаването им [1, 2, 5, 8, 9 и др.]. Тези тематики се съдържат и в актуалните учебници за средното училище [3, 4, 6, 7 и др.].

В настоящата разработка ще представим някои идеи и подходи за създаване на методически инструментариуми за преподаване на програмиране.

2. Преподаване на програмиране с едновременното използване на езиците C#, Java и Python

Тук ще покажем една техника на преподаване на програмиране чрез паралелно използване на три програмни езика.

Задача 1. Да се напише програма, която при въвеждане на целочислена стойност за x , извежда подходящо съобщение за това дали числото е по-голямо или по-малко и равно от 10.

C#	Java	Python
<pre>static void Main(string[] args) { Console.WriteLine("Въведете число:"); int x = int.Parse(Console.ReadLine()); if (x > 10) Console.WriteLine("Числото е по-голямо от 10"); else Console.WriteLine("Числото е по-малко или равно от 10"); Console.ReadKey(); }</pre>	<pre>import java.util.Scanner; class Main { public static void main(String[] args){ Scanner sc = new Scanner(System.in); System.out.println("Въведете число:"); int x = sc.nextInt(); if (x > 10) System.out.println("Числото е по-голямо от 10"); else System.out.println("Числото е по-малко или равно от 10"); } }</pre>	<pre>print("Въведете число:") x=int(input()) if x>10: print("Числото е по-голямо от 10") else: print("Числото е по-малко или равно от 10")</pre>

Задача 2. Да се напише програма, която извежда целите числа в интервала $[0;10]$, използвайки цикъл с брояч **for**.

C#	Java	Python
<pre>static void Main(string[] args) { for (int i = 0; i <= 10; i++) Console.WriteLine(i); }</pre>	<pre>public static void main(String[] args) { for (int i=0; i<=10; i++) System.out.println(i); }</pre>	<pre>for i in range(11): print(i)</pre>

Задача 3. Да се напише програма, която извежда целите числа от 1 до 10, използвайки цикъл с предусловие и постусловие.

C#	Java	Python
<pre>static void Main(string[] args) { int i = 1; while (i <= 10) { Console.WriteLine(i); i++; } }</pre>	<pre>public static void main(String[] args) { int i=1; while (i<=10) { System.out.println("i="+i); i++; } }</pre>	<pre>i=1 while i<=10: print(i) i+=1</pre>
C#	Java	Python
<pre>static void Main(string[] args) { int i = 1; do { Console.WriteLine(i); i++; } while (i <= 10); }</pre>	<pre>public static void main(String[] args) { int i=1; do { System.out.println(i); i++; } while (i<=10); }</pre>	<pre>i=1 while True: print(i) i+=1 if i > 10: break</pre>

Задача 4. Да се напише програма, която намира най-големия общ делител на две естествени числа по алгоритъма на Евклид с изваждане (докато двете числа са различни, от по-голямото се изважда по-малкото, като в следващата стъпка участват вече резултатът от изваждането и по-малкото число) и да се направи контрол на входните данни срещу некоректно въвеждане (да не могат да се въвеждат отрицателни числа и 0).

C#	Java	Python
<pre>static void Main(string[] args) { int a, b; do { Console.WriteLine("Въведете две естествени числа:"); Console.Write("a="); a = int.Parse(Console.ReadLine()); Console.Write("b="); b = int.Parse(Console.ReadLine()); } while (a <= 0 b <= 0); while (a != b) { if (a > b) a -= b; else b -= a; Console.WriteLine("НОД е " + a); Console.ReadKey(); } }</pre>	<pre>import java.util.Scanner; class Main { public static void main(String[] args) { Scanner sc = new Scanner(System.in); int a,b; do { System.out.println("Въведете две естествени числа:"); System.out.print("a="); a = sc.nextInt(); System.out.print("b="); b = sc.nextInt(); } while (a <= 0 b <= 0); while (a != b) { if (a > b) a -= b; else b -= a; System.out.print("НОД е " + a); } } }</pre>	<pre>print("Въведете две естествени числа:") while True: a=int(input("a=")) b=int(input("b=")) if a > 0 and b > 0: break while a != b: if a > b: a -= b else: b -= a print("НОД е " + str(a))</pre>

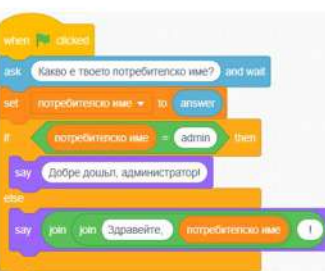
3. Преход от език за блоково програмиране към скриптов език

Тук ще демонстрираме някои примери, с които може да се реализира плавен преход от език с блоково програмиране към два скриптов езика (Python и JavaScript).

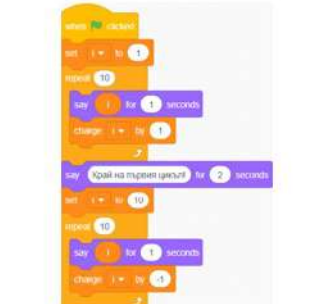
Задача 1. Създайте програма, която пита за вашите име, фамилия и години, след което извежда поздрав: „Здравейте, {Име} {Фамилия} на {Години} години!“.

Scratch	Python
	<pre>print("Какво е твоето име?") name=str(input()) print("Каква е твоята фамилия?") lastName=str(input()) print("На колко години си?") age=str(input()) print("Здравейте, " + name + " " + lastName + " на " + age + " години!")</pre>
	JavaScript
	<pre>name = prompt("Какво е твоето име?"); lastName = prompt("Каква е твоята фамилия?"); age = prompt("На колко години си?"); console.log("Здравей, " + name + " " + lastName + " на " + age + " години!");</pre>

Задача 2. Направете програма, която пита потребителя за неговото потребителско име. Ако потребителското име е „admin“, програмата извежда съобщение „Добре дошъл, администратор!“. За всички останали потребителски имена програмата извежда съобщение „Здравейте, {потребителско име}!“

Scratch	Python
	<pre>print("Какво е твоето потребителско име?") username=str(input()) if username=="admin": print("Добре дошъл, администратор!") else: print("Здравейте, " + username + "!")</pre>
	JavaScript
	<pre>username = prompt("Какво е твоето потребителско име?"); if (username == "admin") { console.log("Добре дошъл, администратор!"); } else { console.log("Здравейте, " + username + "!"); }</pre>

Задача 3. Направете програма, която отпечатва числата от 1 до 10, след това отпечатайте числата в обратен ред от 10 до 1.

Scratch	Python
	<pre>for i in range(1,11,1): print(i) print("Край на първия цикъл!") for i in range(10,0,-1): print(i)</pre>
	JavaScript
	<pre>for (i = 1; i <= 10; i = i + 1) { console.log(i); } console.log("Край на първия цикъл!"); for (i = 10; i >= 1; i = i - 1) { console.log(i); }</pre>

Задача 4. Реализирайте подпрограма, която изчертава правилен многоъгълник по зададени брой страни и дължина на страната.

Scratch	Python
	<pre>import turtle pen=turtle.Turtle() def draw_Polygon(sides,sideLength): for i in range(sides): pen.forward(sideLength) pen.left(360/sides) print("Брой страни:") a=int(input()) print("Дължина на страната:") b=int(input()) draw_Polygon(a,b)</pre>
	JavaScript
	<pre>function drawPolygon(sides, sideLength) { for (let i = 0; i < sides; i++) { turtle.forward(sideLength); turtle.right(360 / sides); } } a = parseInt(prompt("Брой страни:")); b = parseInt(prompt("Дължина на страната:")); drawPolygon(a, b);</pre>

4. Използване на вградени средства за програмиране в среда на Excel

Тук ще демонстрираме примери, с които се програмират някои базови алгоритми чрез използване на Visual Basic for Applications (VBA) в Excel:

- размяна на стойности на две променливи;
- подреждане на три елемента по големина;
- намиране на минимален и максимален елемент на редица от числа.

Задача 1. Със средствата на Excel и VBA да се реализира дейността по размяна на стойности на две клетки (с целочислени данни) в работен лист, използвайки алгоритъма за размяна на стойности на две променливи.

	A	B	C	D	
1	Размяна на стойности на две клетки				
2					
3		26			
4					
5					
6					
7					
8		12			
9					

Фигура 1. Примерен модел на зад. 1 в среда на Excel

Програмен код на VBA

```
Public Sub razmiana_AB()
    Dim A, B, P As Integer
    If Sheet_zad1.Range("B3").Value <> "" And Sheet_zad1.Range("B8").Value <> "" And _
        IsNumeric(Sheet_zad1.Range("B3")) And IsNumeric(Sheet_zad1.Range("B8")) Then
        A = Int(Sheet_zad1.Range("B3").Value)
        B = Int(Sheet_zad1.Range("B8").Value)
        P = A
        A = B
        B = P
        Sheet_zad1.Range("B3").Value = A
        Sheet_zad1.Range("B8").Value = B
    Else
        MsgBox "Непълни или некоректни данни!", vbCritical, "Грешка"
    End If
End Sub
```

Задача 2. Със средствата на Excel и VBA да се реализира дейността по подреждане на три цели числа в нарастващ ред (*разположени в последователни една под друга клетки в работен лист*), използвайки алгоритъма за подреждане на три елемента по големина. За клетките, съдържащи входните данни да се приложи условно форматиране, както и валидиране на данни. За автоматизиране на попълването на входни числови данни да се използва случайно генериране на цели числа в интервала [1,150].



Фигура 2. Примерен модел на зад. 2 в среда на Excel

Програмен код на VBA

Програмен код на VBA	
<pre>Public Sub razmiana(A, B) Dim p As Integer p = A A = B B = p End Sub Public Sub sort_3chisla_SmalltoLarge() Dim A, B, C As Integer A = Val(Sheet_zad2.Range("B3").Value) B = Val(Sheet_zad2.Range("B4").Value) C = Val(Sheet_zad2.Range("B5").Value) If A > B Then razmiana A, B If A > C Then razmiana A, C If B > C Then razmiana B, C Sheet_zad2.Range("B3").Value = A Sheet_zad2.Range("B4").Value = B Sheet_zad2.Range("B5").Value = C End Sub</pre>	<pre>Public Sub sort_3chisla_LargetoSmall() Dim A, B, C As Integer A = Val(Sheet_zad2.Range("B3").Value) B = Val(Sheet_zad2.Range("B4").Value) C = Val(Sheet_zad2.Range("B5").Value) If A < B Then razmiana A, B If A < C Then razmiana A, C If B < C Then razmiana B, C Sheet_zad2.Range("B3").Value = A Sheet_zad2.Range("B4").Value = B Sheet_zad2.Range("B5").Value = C End Sub Public Sub random_sort_3chisla() For i = 1 To 3 rndvalue = Int(150 * Rnd + 1) Sheet_zad2.Cells(2 + i, 2).Value = rndvalue Next i End Sub</pre>

Задача 3. Със средствата на Excel и VBA да се реализира дейността по намиране на минимален и максимален елемент в редица от цели числа (включваща от 3 до 10 елемента), въведени в дадена област от клетки в работен лист, като за целта се използва алгоритъм за намиране на минимален/максимален елемент от повече от три елемента. За автоматизиране на дейността по автоматично попълване на входните числови данни да се използва техника за случайно генериране на цели числа в интервала [1,100].



Фигура 3. Примерен модел на зад. 3 в среда на Excel

Програмен код на VBA

```
Public Sub MinMax_RedicaChisla()
    Dim min, max, br As Integer
    Sheet_zad3.Range("F12").Formula = "=COUNT(B4:B13)"
    br = Sheet_zad3.Range("F12").Value
    If br >= 3 Then
        min = Sheet_zad3.Range("B4").Value
        max = Sheet_zad3.Range("B4").Value
        For i = 2 To br
            If Sheet_zad3.Cells(i + 3, 2).Value < min Then min = Sheet_zad3.Cells(i + 3, 2).Value
            If Sheet_zad3.Cells(i + 3, 2).Value > max Then max = Sheet_zad3.Cells(i + 3, 2).Value
        Next i
        Sheet_zad3.Range("F8").Value = min
        Sheet_zad3.Range("F10").Value = max
    Else
        MsgBox "Непълни данни!"
    End If
End Sub

Public Sub random_fill_N_chisla()
    Dim n As Integer
    clear_data_zad3
    'генериране на случайно число в интервала [a,b]
    'Int((b - a + 1) * Rnd + a)
    N = Int(8 * Rnd + 3)
    For i = 1 To N
        Sheet_zad3.Cells(i + 3, 2).Value = Int(100 * Rnd + 1)
    Next i
End Sub

Public Sub clear_data_zad3()
    Sheet_zad3.Range("B4:B13,F8,F10,F12").ClearContents
End Sub
```

5. Заключение

В настоящата статия разгледахме някои основни елементи от подготовката на бъдещите учители в областта на програмирането. Беше извършен анализ на учебните курсове, свързани с програмиране. Предс-

тавен е методически инструментариум за преподаване на програмиране, засягащ три основни аспекта:

- инструментариум за преподаване на програмиране с едновременното използване на три програмни езика (C#, Java и Python);
- инструментариум за реализиране на плавен преход от език за блоково програмиране към програмиране чрез използване на два скриптов езика (Python и JavaScript);
- инструментариум, свързан с използване на вградени средства за програмиране в среда на Excel, предоставящ възможности за комплексно развитие на алгоритмични знания и дигитални умения.

Настоящата разработка търпи развитие в разработване на допълнителен инструментариум за преподаване на програмиране.

Литература

- [1] С. Анева, Е. Тодорова, Възможности за развитие на алгоритмични умения на учениците в обучението по предмета „Компютърно моделиране и информационни технологии“ в прогимназията, *Сборник доклади на Юбилейна международна научна конференция „Компютърни технологии и приложения“*, Пампорово, 2021, 37–46, ISBN: 978-619-202-702-5.
- [2] С. Анева, Е. Тодорова, Развитие на дигитални и алгоритмични умения на учениците чрез използване на Visual Basic for Applications, *Математика и информатика*, година LXVII, кн. 3, 2024, 314–335, ISSN:1310-2230 (Print), DOI: <https://doi.org/10.53656/math2024-3-6-dev>.
- [3] К. Гъров, Р. Папанчева, С. Анева, Т. Глушкова, Г. Стоицов, Е. Тодорова, И. Велчева, Д. Данаилов, *Компютърно моделиране и информационни технологии за 6. клас за задължителна подготовка*, Клет България, София, 2022, ISBN: 978-619-7669-03-9.
- [4] К. Гъров, Р. Папанчева, С. Анева, Т. Глушкова, Г. Стоицов, Е. Тодорова, И. Велчева, Д. Данаилов, *Компютърно моделиране и информационни технологии за 7. клас за задължителна подготовка*, Клет България, София, 2023, ISBN: 978-619-7669-07-7.
- [5] Х. Крушков, *Програмиране на C#*, Коала Прес, Пловдив, 2017, ISBN: 978-619-7134-44-5.

- [6] К. Манев, Н. Манева, В. Христова, *Информатика 8. клас общообразователна подготовка*, Клет България, София, 2024, ISBN: 978-619-7669-17-6.
- [7] Г. Момчева, Т. Глушкова, Р. Маринова, *Информатика учебник за 8. клас*, Клет България, София, 2024, ISBN: 978-954-344-783-1.
- [8] Е. Тодорова, О. Челик, Основни характеристики на обучението по програмиране в българското училище, *Е-списание „Образование и развитие“*, бр. 14, 2024, стр. 94–103, ISSN: 2603-3577.
- [9] Е. Тодорова, С. Цанкова, Един методически подход за изучаване на JavaScript в училище, *Е-списание „Образование и развитие“*, бр. 14, 2024, стр. 104–115, ISSN: 2603-3577.

Коста Гъров¹, Стефка Анева¹, Елена Тодорова¹,

¹ Пловдивски университет „Паисий Хилендарски“

Факултет по математика и информатика,

бул. „България“ № 236, Пловдив, България

Автор за кореспонденция: kosgar@uni-plovdiv.bg

TEACHING PROGRAMMING IN TEACHER TRAINING

Kosta Garov, Stefka Aneva, Elena Todorova

Abstract. *This current paper is dedicated to the teaching of Programming in teacher training at the Faculty of Mathematics and Informatics at the University of Plovdiv “Paisii Hilendarski”. The programming courses take a leading role in the curricula for the bachelor programs “Mathematics, Informatics and Information Technologies” and “Information Technologies, Mathematics and Education Management”, the master programs “Teaching Informatics and Information Technologies at school” and “Pedagogy of teaching Informatics and Information Technologies” and the qualification “Teacher in Informatics and Information Technologies”. In this work are analyzed current curricula in Informatics for the corresponding majors and master programs. Several ideas are presented for the methodology of teaching Programming and Computer Modeling, aimed at the middle schools. Various approaches are presented for creating the elements of the methodical apparatus.*

Key words: Programming, Informatics, Computer Modeling, Teaching Methodology.