

РАЗВИТИЕ НА ДИГИТАЛНИ И АЛГОРИТМИЧНИ УМЕНИЯ НА УЧЕНИЦИТЕ ЧРЕЗ ИЗПОЛЗВАНЕ НА VISUAL BASIC FOR APPLICATIONS

Стефка Анева, Елена Тодорова

Пловдивски университет „Паисий Хилендарски“ (България)

Резюме. В настоящата статия се представя един подход за интегрирано вътрешнопредметно взаимодействие и креативно използване на формираните дигитални умения на учениците за обработка на таблични данни, както и техните умения в областта на програмирането при реализацията на познати за тях алгоритми от учебното съдържание по компютърно моделиране и информационни технологии в 5. клас в нов контекст и среда за програмиране. Представени са шест задачи и техните решения, реализирани с помощта на вградените средства за програмиране в Excel. С тези задачи се цели използването на линейни, разклонени и циклични алгоритми за автоматизиране на изпълнението на следните дейности – размяна на стойности на две клетки, намиране на минимална и максимална стойност на три или повече клетки в работен лист и подреждане на три или повече елемента по големина.

Ключови думи: компютърно моделиране; информационни технологии; дигитални умения; дигитална креативност; алгоритмични умения; алгоритми; методика на обучение; електронни таблици; Visual Basic for Applications

1. Увод

Съвременното образование е необходимо да осигури подходящи условия и възможности за развитие и усъвършенстване на знанията и уменията на учениците в сферата на информационните технологии, компютърните науки и програмирането през всички образователни степени. Подготовката на качествени специалисти в областта на информационните и комуникационните технологии и програмирането трябва да бъде системен и непрекъснат процес и да започва още от ранна детска възраст (Aneva & Todorova 2021).

Придобиването на дигитални и алгоритмични умения в българското образование започва в началното училище и се надгражда и развива във

всеки следващ етап на обучение (Papancheva et al. 2018; Manev et al. 2017; Dureva 2003, Slavova & Garov 2019).

Съществен аспект в учебния процес при компетентностно ориентирано образование се явява необходимостта от осигуряване на реален практически контекст за целите на обучението и разработване на учебните задачи (Garov 2010) по начин, който стимулира творческото прилагане на натрупаните знания и умения и формиране на „меки“ умения (инициативност, креативност, критично мислене, емоционалната интелигентност, адаптивност, работа в екип, вземане на решения и др.).

Според (Kozhuharova 2020) креативността може да бъде обособена като:

- нестандартен поглед към проблемите и търсене на нови гледни точки към тях;
- преформулиране на вече известни проблеми с цел намиране на оригинални идеи за тяхното разрешаване;
- отклоняване на мисленето от традиционните схеми;
- способност да се раждат необикновени идеи, представящи иновативни решения за бързо разрешаване на различни проблемни ситуации.

Обучението по компютърно моделиране и информационни технологии в прогимназиален етап¹ е насочено към:

- овладяване на базисни знания, умения и отношения, свързани с учебната дисциплина, които са задължителна част от техническата грамотност на обучаемите и са в съответствие с изискванията на съвременното информационно общество;
- познаване на възможностите на съвременните информационни технологии за обработване на различен тип информация и творческото използване на тези знания при решаване на реални практически проблеми с различна степен на сложност;
- надграждане и развитие на знанията на учениците за използване на блоков език за програмиране, формиране на знания и умения за използване на скриптов текстов език и създаване на интерактивни приложения с него;
- изграждане на дигитални компетентности на ученика и приложението им в различни предметни области;
- формиране и развитие на ключови компетентности, чрез които учениците развиват своите способности за творческо и креативно мислене.

2. Реализации на някои алгоритми чрез прилагане на формираните дигитални умения на учениците за работа с електронни таблици и използване на вградените средства за програмиране в среда на Excel

Обучението по компютърно моделиране и информационни технологии в прогимназиален етап предлага благоприятна среда, чрез която:

- учениците формират и развиват дигитални умения за работа с различна по тип информация чрез използване на различни приложни програми;
- учениците развиват и усъвършенстват своите алгоритмични знания и умения в сферата на програмирането, като използват блоков и скриптов текстов език за програмиране за реализация на интерактивни учебни задачи и проекти;
- същевременно да се провокират знанията, уменията, въображението и креативността на учениците чрез творческо интегрирано прилагане на формираните дигитални и алгоритмични умения. Това се постига чрез използване на богатите възможности на информационните технологии за извършване на разнообразни дейности, реализиране на различни проекти, за експериментиране на варианти, за търсене на нови подходи, стратегии и решения.

В настоящата работа представяме примерен подход за интегрирано вътрешнопредметно взаимодействие и креативно използване на формираните дигитални умения на учениците за обработка на таблични данни и техните алгоритмични умения за използване на линейни, разклонени и циклични конструкции при реализацията на познати за тях алгоритми (изучавани в 5. клас) в нов контекст и среда за програмиране. За целта ще използваме възможностите на Excel и вградените средства за програмиране в тази среда на езика Visual Basic for Applications (VBA), и ще разгледаме примерни реализации със средствата на Excel и VBA на следните дейности:

- размяна на стойности на две променливи;
- намиране на минимален/максимален от три или повече елемента;
- подреждане на три или повече елемента по големина.

С езика VBA могат да бъдат автоматизирани редица дейности и процеси, ръчното изпълнение на които само със средствата на приложението (в случая Excel) би отнело повече време или би предизвикало трудност, както и да бъдат разширени функционалностите на приложенията на MS Office. Създадените подпрограми (макродефиниции, макроси) чрез средствата на VBA могат да управляват автоматично изпълнението на

определена последователност от действия, като по този начин се увеличи и ефективността на работа със съответното приложение.

Предложеният подход би могъл да бъде приложен успешно в обучението по КМИТ в 6. и 7. клас (разширена или допълнителна подготовка), както и за организиране и провеждане на извънкласни дейности и занимания по интереси.

2.1. Реализация на алгоритъма за размяна на стойности на две променливи чрез средствата на Excel и VBA

Най-напред би могло да се дискутира с учениците самата проблемна ситуация: *как да бъдат разменени стойностите на две клетки в Excel?* Възможните варианти за разрешаване на проблема могат да бъдат обсъдени в два аспекта.

От гледна точка на техните знания и формирани умения за работа с електронни таблици – би трябвало да се използва помощна клетка, за да се осъществи размяната.

От гледна точка на техните знания и умения за прилагане на алгоритъма за размяна на стойности на две променливи, формиран в 5. клас при изучаване на темата „Компютърно моделиране“ – в случая стойностите на двете клетки от даден работен лист в Excel, могат да се присвоят като стойности на две променливи, след което да се приложи алгоритъмът за размяна на стойностите на тези променливи и полученият резултат да се визуализира отново в двете клетки в Excel.

Тук ще представим една задача, илюстрираща автоматизирането на дейностите по размяна на стойности на две клетки в Excel.

Задача 1. Със средствата на Excel и VBA реализирайте дейността по размяна на стойностите на две клетки в Excel (цели числа), като приложите алгоритъма за размяна на стойности на две променливи.

Решение: Най-напред в работен лист в Excel е необходимо да се реализира конкретният модел на задачата по посочения примерен образец на фиг. 1. За въвеждане на двете стойности ще се използват конкретни клетки от работен лист в Excel – в случая В3 и Г3.

Следващият етап от решението на задачата е реализирането на необходимите



Фигура 1. Примерен модел на задача 1

функционалности с помощта на средата на Visual Basic Editor и езика VBA във вид на конкретни макроси – за размяна на стойности на двете клетки и за подготовка за ново въвеждане.

С помощта на макроса **swap_AB_zad1** се реализира дейността по размяна на стойностите на две избрани клетки (в случая B3 и G3). Работният лист в Excel, в който е реализиран моделът на задачата, е препоръчително да бъде именуван като обект във VBA по подходящ начин – в случая е именуван като **Sheet_zad1**.

Листинг 1. Реализация на алгоритъма за размяна на стойностите на клетки B3 и G3 с проверка за пълнота и коректност

```
Public Sub swap_AB_zad1()  
    Dim A, B, swap As Integer  
    If Sheet_zad1.Range("B3")<>" " And Sheet_zad1.Range("G3") _  
        <> " " And IsNumeric(Sheet_zad1.Range("B3")) And _  
        IsNumeric(Sheet_zad1.Range("G3")) Then  
        A = Int(Sheet_zad1.Range("B3"))  
        B = Int(Sheet_zad1.Range("G3"))  
        swap = A  
        A = B  
        B = swap  
        Sheet_zad1.Range("B3") = A  
        Sheet_zad1.Range("G3") = B  
    Else  
        MsgBox "Непълни или некоректни данни!", vbExclamation  
    End If  
End Sub
```

С помощта на макроса **new_data_swap** може да се автоматизира дейността по изчистване на текущите данни в двете клетки.

Листинг 2. Подготовка за ново въвеждане на данни

```
Sub new_data_swap()  
    Sheet_zad1.Range("B3,G3").ClearContents  
    Sheet_zad1.Range("B3").Select  
End Sub
```

За изпълнението на създадените макроси учениците могат да използват два подхода:

- чрез добавяне на бутон (Developer/Insert/Button (Form Control)), който да се свърже с изпълнението на даден макрос;
- чрез използване на графични форми или изображения, които да бъдат свързани с даден макрос – това се постига чрез изпълнение на командата **Assign Macro . . .**, достъпна от контекстното меню за съответния обект, и избор на макрос, който да бъде свързан с обекта.

И в двата случая, при щракване върху съответния обект, който е свързан с определен макрос, ще се стартира изпълнението на съответната макродефиниция. За изпълнението на макросите, реализиращи функционалностите в Задача 1, са използвани графични форми, които се свързват към съответните макроси.

2.2. Реализация на алгоритъма за намиране на минимален или максимален от три или повече елемента

Най-напред би могло да се дискутира с учениците проблемната ситуация: *Как да се определи минималният и максималният елемент за произволен брой числови данни, въведени в последователни клетки в работен лист на Excel и разположени в един стълб?* Възможните варианти за разрешаване на проблема могат да бъдат обсъдени в два аспекта.

От гледна точка на техните знания и формирани умения за работа с електронни таблици в 6. клас – ако броят на клетките с въведени данни е предварително известен. В този случай учениците биха използвали вградените функции **MIN** и **MAX** за определяне на минималния и максималния елемент в определени клетки, в които трябва да се визуализира резултатът.

От гледна точка на техните знания и умения, формирани в 5. и 6. клас при изучаване на темата „Компютърно моделиране“, относно прилагане на алгоритми за намиране на минимален/максимален елемент от три или повече елемента, реализирани чрез средствата на език с блоково програмиране и скриптов текстов език. В този случай могат да бъдат формулирани две задачи, илюстриращи прилагането на съответните алгоритми за намиране минимален/максимален елемент най-напред за три елемента (т.е. предварително известен брой на елементите, които се анализират), а след това и за произволен брой. При реализирането на двете задачи се предполага, че необходимите входни числови данни (цели числа) са коректно въведени.

Задача 2. Със средствата на Excel и VBA реализирайте дейността по намиране минимален/максимален елемент за въведени в три клетки в работен лист на Excel данни, като за целта приложите познатия ви алгоритъм за намиране на минимален/максимален от три елемента. За автоматизиране на дейността по въвеждане на данни за трите клетки

реализирайте възможност за автоматично попълване чрез използване на случайно пораждање на числа в интервала [1,100].

Решение: В работен лист на Excel е необходимо да се реализира конкретният модел на задачата, по посочения примерен образец на фиг. 2. В случая е акцентирано само на дейността по намиране на минималния елемент. Аналогично може да се подходи и при намирането на максималния елемент и да се разшири функционалността на решението. За въвеждане на трите стойности ще се използват последователни клетки от работен лист в Excel – в случая са избрани клетките B6, B7 и B8, но клетките могат да бъдат и произволно избрани. След определяне на минималния елемент неговата стойност ще се визуализира в клетка D3. Работният лист в Excel, в който е реализиран моделът на задачата, е обект на VBA с име **Sheet_min**.

Въвеждането на числови данни в трите клетки може да се реализира чрез ръчно въвеждане или със случайно пораждање на числови стойности в интервала [1, 100].

Следващият етап от решението на задачата е реализирането на някои основни функционалности с макроси на VBA – за намиране на минималния елемент, за случайно пораждање на стойностите за трите клетки и за подготовка за ново въвеждане. Допълнително към задачата може да се реализира и проверка за пълнота на въведени числови данни в трите клетки (в случай, че данните се въвеждат ръчно) и след това тези данни да бъдат анализирани за определяне на минимален/максимален елемент.

При реализация на алгоритъма за търсене на минимален елемент от три въведени стойности на учениците могат да бъдат представени два примерни варианта за решение.

Първи вариант – чрез използване на три променливи (A, B и C), в които да се извлекат въведените в клетките B6, B7 и B8 стойности и да се определи минималният елемент чрез прилагане на вложени условни оператори. В процеса на търсене на минималния елемент се използва помощна променлива **min**. Този вариант е приложим както за последователни, така и за произволно избрани клетки в работния лист.

Втори вариант – без използване на отделни променливи за извличане на стойностите на трите клетки и прилагане на оператор за цикъл **For** при обхождане на отделните клетки за определяне на минималния еле-



Фигура 2. Примерен модел на задача 2

мент. В процеса на търсене на минималния елемент се използва помощна променлива **min**, която първоначално приема стойността на първата клетка (в случая B6). След това с помощта на оператора за цикъл **For** се реализира директно обхождане на останалите две клетки и сравнение с текущата стойност на променливата **min** към момента. В случай, че някоя от тези две клетки има стойност, която е по-малка от текущата стойност на **min**, то стойността на тази клетка се присвоява като стойност на променливата **min**. Този вариант е приложим само в случай, че са избрани последователно разположени (в стълб или в ред) клетки на работния лист. Представеният алгоритъм лесно може да се доразвие за намиране на минимален елемент на повече от три елемента.

Листинг 3. Първи вариант за реализация на алгоритъма за търсене на минимален на три елемента

```
Public Sub min_el_3elementa()  
    Dim A, B, C, min As Integer  
    A = Val(Sheet_min.Range("B6").Value)  
    B = Val(Sheet_min.Range("B7").Value)  
    C = Val(Sheet_min.Range("B8").Value)  
    If A < B Then  
        If A < C Then min = A Else min = C  
    Else  
        If B < C Then min = B Else min = C  
    End If  
    Sheet_min.Range("D3").Value = min  
End Sub
```

Листинг 4. Втори вариант за реализация на алгоритъма за търсене на минимален на три елемента (при последователно разположени в даден стълб клетки)

```
Public Sub min_el_v2()  
    Dim br, min, i As Integer  
    br = 3  
    min = Sheet_min.Range("B6").Value  
    For i = 2 To br  
        If min > Sheet_min.Cells(i + 5, 2).Value Then _  
            min = Sheet_min.Cells(i + 5, 2).Value  
    Next i  
    Sheet_min.Range("D3").Value = min  
End Sub
```

Забележка. Чрез `cells(i+5,2)` в цикъла `for`, с управляваща променлива `i`, се достъпват последователно клетките B7 и B8, като `i+5` е номерът на реда (7 при `i = 2`, 8 при `i = br`), а 2 е номерът на колоната B за съответна клетка.

Аналогично може да се подходи и при определяне на максималния елемент за въведените стойности в трите клетки от работния лист в Excel.

С макроса `clear_data_min` изчистваме съдържанието на клетките B6, B7, B8 и D3, за да ги подготвим за въвеждане на нови стойности.

Листинг 5. Реализиране на дейността, свързана с подготовка за ново въвеждане на данни

```
Sub clear_data_min()  
    Sheet_min.Range("B6:B8").ClearContents  
    Sheet_min.Range("D3").ClearContents  
    Sheet_min.Range("B6").Select  
End Sub
```

Листинг 6. Реализиране на функционалността за автоматично попълване на входни данни чрез случайно пораждање на стойностите за трите клетки в интервала [1,100]

```
Public Sub random_min_3chisla()  
    clear_data_min  
    For i = 1 To 3  
        rndvalue = Int(100 * Rnd + 1)  
        Sheet_min.Cells(5 + i, 2).Value = rndvalue  
    Next i  
End Sub
```

Задача 3. Със средствата на Excel и VBA реализирайте намирането на минимален и максимален елемент на въведени числови данни в N клетки ($N \in [3, 20]$), разположени в един стълб на работния лист, като приложите алгоритъма за намиране на минимален/максимален за повече от три елемента. За автоматизиране на дейността по въвеждане на данни за отделните клетки реализирайте функционалност за автоматично попълване чрез използване на случайно пораждање на числа в интервала [1,100].

Решение: В работен лист на Excel е необходимо да се реализира конкретният модел на задачата по посочения примерен образец на фиг. 3. За въвеждане на данни ще се използва стълб A в работен лист на

Excel. В зависимост от избрания брой N на елементите, чрез случайно пораждаване в интервала $[3,20]$, се извършва автоматизирано попълване на съответния брой елементи в стълб А. След определяне на минималния и максималния елемент съответните стойности ще се визуализират в клетки D6 и D9. Работният лист в Excel, в който е реализиран моделът на задачата, е обект във VBA с име Sheet_MinMax_N.

	A	B	C	D	E
	Елементи	Определяне на минимален и максимален елемент от произволен брой елементи			
1					
2					
3			Въведен брой елементи		
4					
5					
6			Минимален елемент		
7					
8					
9			Максимален елемент		
10					
11					
12					
13			Намиране на минимален и максимален елемент		
14					
15					
16					
17			Случайно генериране на N на брой стойности ($N \in [3; 20]$)		
18					
19					
20					
21					

Фигура 3. Примерен модел на задача 3

Следващият етап от решението на задачата е реализирането на необходимите функционалности с помощта на VBA и създаването на съответните макроси, автоматизиращи намиране на минималния и максималния елемент, случайно пораждаване на N ($N \in [3, 20]$) елемента, като съответните числови данни за всяка клетка да са в интервала $[1,100]$ и подготовка за ново въвеждане.

Листинг 7. Реализация на алгоритъма за търсене на минимален и максимален на произволен брой елементи

```
Public Sub min_max_proizvolen_br_el()  
    Dim br, min, max As Integer  
    Sheet_MinMax_N.Range("D3").Formula = "=COUNT(A:A)"  
    br = Sheet_MinMax_N.Range("D3").Value  
    min = Sheet_MinMax_N.Range("A2").Value  
    max = Sheet_MinMax_N.Range("A2").Value  
    For i = 2 To br  
        If min > Sheet_MinMax_N.Cells(i + 1, 1).Value _  
            Then min = Sheet_MinMax_N.Cells(i + 1, 1).Value  
        If max < Sheet_MinMax_N.Cells(i + 1, 1).Value _  
            Then max = Sheet_MinMax_N.Cells(i + 1, 1).Value  
    Next i  
    Sheet_MinMax_N.Range("D6").Value = min  
    Sheet_MinMax_N.Range("D9").Value = max  
End Sub
```

Листинг 8. Реализиране на дейността, свързана с подготовка за ново въвеждане на данни

```
Sub clear_data()  
    Sheet_MinMax_N.Range("A2:A21").ClearContents  
    Sheet_MinMax_N.Range("D3,D6,D9").ClearContents  
End Sub
```

Листинг 9. Реализиране на функционалността за автоматично попълване на входни данни чрез случайно пораждаване в стълб А на стойностите за N клетки ($N \in [3, 20]$) в интервала $[1, 100]$

```
Public Sub random_N_chisla()  
    Dim N As Integer  
    clear_data  
    ' пораждаване на случайно число в интервала [a,b]  
    ' Int((b - a + 1) * Rnd + a)  
    ' пораждаване на случайно число в интервала [3,20]  
    N = Int(18 * Rnd + 3)  
    Sheet_MinMax_N.Range("D3").Value = N  
    For i = 1 To N  
        rndvalue = Int(100 * Rnd + 1)
```

```
Sheet_MinMax_N.Cells(1 + i, 1).Value = rndvalue  
Next i  
End Sub
```

Забележка. В процедурата `random_N_chisla` сме включили извикване на процедурата `clear_data` от Листинг 8, която изчиства данните в областта A2:A21, както и в клетки D3, D6 и D9, за да ги подготви за ново въвеждане. Към решението на задачата може да бъде добавен бутон, който да се свърже с `clear_data`, за да се изчистят старите данни, а новите да се въведат ръчно от потребителя.

2.3. Реализация на алгоритъма за подреждане на три и повече елемента по големина

Най-напред би могло да се дискутира с учениците самата проблемна ситуация: *как да се извърши подреждане на произволен брой числови данни, въведени в последователни клетки в работен лист на Excel и разположени в един стълб?* Възможните варианти за разрешаване на проблема могат да бъдат обсъдени в два аспекта

От гледна точка на техните знания и формиранни умения за работа с електронни таблици в 7. клас – ако областта с въведени числови данни е предварително известна, в случая биха използвали команда **Sort** за извършване на дейността по подреждане на числови данни в Excel и подреждане на стойностите в съответните клетки по големина. Биха възникнали някои проблеми, в случай че до стълб с въведени данни, подлежащи на подреждане, има данни (в следващия или предходния стълб, намиращи се на редовете, в които са разположени и числовите данни за подреждане), които не са смислово свързани и не трябва да бъдат обхванати при разпознаване на областта, подлежаща на подреждане;

От гледна точка на техните знания и умения, формиранни в 5., 6. и 7. клас при изучаване на темата „Компютърно моделиране“, относно реализация чрез скриптов текстов език на алгоритми за подреждане на три и повече елемента по големина чрез използване на разклонен или цикличен алгоритъм – в случая могат да бъдат представени три задачи, илюстриращи прилагането на съответните алгоритми за:

- подреждане на три елемента по големина – в случая се използват три променливи A, B и C;
- подреждане на предварително зададен брой елементи – в случая се използва цикъл, чрез който се извършва обхождане на клетките и извличане на стойностите им в едномерен масив;
- подреждане на произволен брой елементи – в този случай най-напред трябва да се намери броят на елементите, след което задачата може да се сведе до втория вариант.

Задача 4. Със средствата на Excel и VBA реализирайте дейността по подреждане на три елемента по големина по въведени данни в три клетки в работен лист на Excel, като за целта приложите познатия алгоритъм за подреждане на три елемента по големина. За автоматизиране на дейността по въвеждане на данни за трите клетки реализирайте функционалност за автоматично попълване чрез използване на случайно пораждаване на числа в интервала [1,100].

Решение: В работен лист на Excel е необходимо да се реализира конкретният модел на задачата по посочения примерен образец на фиг. 4. За въвеждане на трите стойности ще се използват последователните клетки B4, B5 и B6 на работния лист. След извършване на дейността по подреждане, крайният резултат ще се представи в същите клетки. Работният лист в Excel, в който е реализиран моделът на задачата, е обект във VBA с име `Sheet_sort`.

Следващият етап от решението на задачата е реализирането на необходимите функционалности чрез реализация на VBA макроси. Ще представим само реализацията на макроса, автоматизиращ дейността по подреждане на числовите данни в трите клетки по големина, чрез използване на алгоритъма за подреждане на три елемента по големина (в случая във възходящ ред). Останалите дейности, свързани с въвеждане на нови данни чрез случайно пораждаване на съответните стойности за трите

	A	B	C	D	E
1	Алгоритъм за подреждане на три числа по големина				
2					
3		Стойности		Подреди във възходящ ред	
4		29			
5		31			
6		78			
7				Случайно генериране на трите стойности	
8					
9					

Фигура 4. Примерен модел на задача 4

Листинг 10.

```

1 print("Въведете три числа: ")
2 a = int(input())
3 b = int(input())
4 c = int(input())
5 if a>b:
6     swap = a
7     a = b
8     b = swap
9 if a>c:
10    swap = a
11    a = c
12    c = swap
13 if b>c:
14    swap = b
15    b = c
16    c = swap
17 print("Подреждане: ",a,b,c)

```

клетки, могат да бъдат реализирани както при Задача 2. Допълнително към задачата може да се реализира и проверка за пълнота на въведени числови данни в трите клетки (в случай, че данните се въвеждат ръчно) и след това за тези данни да бъде извършено подреждане по големина.

При реализацията на дейността по подреждане може най-напред да бъде направен кратък анализ на примерна програма на Python, представяща реализация на алгоритъма за подреждане на три числа по големина (Листинг 10).

Целта е учениците да разграничат и отделят повтарящи се компоненти на задачата, свързани с приложението на алгоритъма за размяна на стойности на две променливи или дейността по подреждане на два елемента във възходящ ред. В резултат, при реализацията на алгоритъма чрез Python, може да бъде дефинирана нова подпрограма (функция), извършваща някоя от тези функционалности (Todorova et al. 2021).

Листинг 11. Реализация на алгоритъма за подреждане
на три числа по големина чрез VBA

```
Public Sub sort_3chisla()  
    Dim A, B, C, swap As Integer  
    A = Sheet_sort.Range("B4").Value  
    B = Sheet_sort.Range("B5").Value  
    C = Sheet_sort.Range("B6").Value  
    If A > B Then  
        swap = A  
        A = B  
        B = swap  
    End If  
    If A > C Then  
        swap = A  
        A = C  
        C = swap  
    End If  
    If B > C Then  
        swap = B  
        B = C  
        C = swap  
    End If  
    Sheet_sort.Range("B4").Value = A  
    Sheet_sort.Range("B5").Value = B  
    Sheet_sort.Range("B6").Value = C  
End Sub
```

Листинг 12. Реализация на алгоритъма за подреждане на три числа в нарастващ ред чрез VBA с подпрограма за размяна на стойности на две променливи

```
Public Sub razmeni(A, B)
    Dim swap As Integer
    swap = A
    A = B
    B = swap
End Sub
Public Sub sort_3chisla_v2()
    Dim A, B, C As Integer
    A = Sheet_sort.Range("B4").Value
    B = Sheet_sort.Range("B5").Value
    C = Sheet_sort.Range("B6").Value
    If A > B Then
        razmeni A, B
    End If
    If A > C Then
        razmeni A, C
    End If
    If B > C Then
        razmeni B, C
    End If
    Sheet_sort.Range("B4").Value = A
    Sheet_sort.Range("B5").Value = B
    Sheet_sort.Range("B6").Value = C
End Sub
```

Листинг 13. Реализация на алгоритъма за подреждане на три числа в нарастващ ред чрез VBA с подпрограма за подреждане на два елемента в нарастващ ред

```
Public Sub sort_2chisla(A, B)
    Dim swap As Integer
    If A > B Then
        swap = A
        A = B
        B = swap
    End If
End Sub
```

```
Public Sub sort_3chisla_v3()
    Dim A, B, C As Integer
    A = Sheet_sort.Range("B4").Value
    B = Sheet_sort.Range("B5").Value
    C = Sheet_sort.Range("B6").Value
    sort_2chisla A, B
    sort_2chisla A, C
    sort_2chisla B, C
    Sheet_sort.Range("B4").Value = A
    Sheet_sort.Range("B5").Value = B
    Sheet_sort.Range("B6").Value = C
End Sub
```

Задача 5. Със средствата на Excel и VBA реализирайте подреждане по големина на числата от пет клетки в работен лист на Excel.

Решение: В работен лист на Excel е необходимо да се реализира конкретният модел на задачата по посочения примерен образец на фиг. 5. За въвеждане на петте стойности ще се използват последователно разположени клетки от работен лист в Excel – в случая това е областта B4:B8. След подреждането, резултатът ще се визуализира в същата област. Работният лист в Excel, в който е реализиран моделът на задачата, е обект във VBA с име Sheet_sort_5chisla.

Следващият етап от решението на задачата е реализирането на необходимите функционалности с помощта на VBA и създаването на съответните макроси, автоматизиращи следните дейности – подреждане на числовите данни в петте клетки по големина чрез реализиране на алгоритъм за подреждане на пет елемента по големина (във възходящ и низходящ ред); въвеждане на нови данни чрез случайно пораждаване на съответните стойности за петте клетки. Допълнително към задачата може да се реализира и проверка за пълнота на въведени числови данни за

	A	B	C	D	E
1	Алгоритъм за подреждане на пет числа по големина				
2					
3		Стойности		Подреди във възходящ ред	
4		53			
5		77		Подреди в низходящ ред	
6		6			
7		60		Случайно генериране на данни	
8		47			
9					
10					

Фигура 5. Примерен модел на задача 5

петте клетки (в случай, че данните се въвеждат ръчно) и след това за тези данни да бъде извършено подреждане по големина.

Листинг 14. Реализация на алгоритъм за подреждане по големина на пет числа с VBA

```
Public Sub sort_5chisla_SmalltoLarge()  
    Dim A(4) As Integer  
    Dim N, p As Integer  
    N = 5  
    For i = 0 To N - 1  
        A(i) = Sheet_sort_5chisla.Cells(4 + i, 2)  
    Next i  
    ' алгоритъм за подреждане по метода на мехурчето  
    For i = 1 To N - 1  
        For j = N - 1 To i Step -1  
            If A(j) < A(j - 1) Then  
                p = A(j)  
                A(j) = A(j - 1)  
                A(j - 1) = p  
            End If  
        Next j  
    Next i  
    For i = 0 To N - 1  
        Sheet_sort_5chisla.Cells(4 + i, 2).Value = A(i)  
    Next i  
End Sub
```

За реализация на низходящо подреждане може да се подходи аналогично.

Листинг 15. Реализиране на функционалността за автоматично попълване на входни данни чрез случайно пораждаване на стойностите за петте клетки от областта B4:B8 в интервала [1,100]

```
Public Sub random_sort_5chisla()  
    For i = 1 To 5  
        rndvalue = Int(100 * Rnd + 1)  
        Sheet_sort_5chisla.Cells(3 + i, 2).Value = rndvalue  
    Next i  
End Sub
```

Задача 6. Чрез средствата на Excel и VBA реализирайте дейността за подреждане по големина на произволен брой цели числа, въведени в последователно разположени в един стълб клетки в работен лист на Excel.

Решение: В работен лист на Excel е необходимо да се реализира конкретният модел на задачата по посочения примерен образец на фиг. 6.

	A	B	C	D	E
1	Елементи	Подреждане по големина на произволен брой въведени елементи			
2					
3		Въведен брой елементи			
4		Подреждане по големина във възходящ ред			
5		Подреждане по големина в низходящ ред			
6		Случайно генериране на N на брой стойности ($N \in [3; 20]$)			
7					
8					
9					
10					
11					

Фигура 6. Примерен модел на задача 6

Листинг 16. Примерна реализация на алгоритъм за подреждане на произволен брой елементи по големина във възходящ ред чрез VBA

```
Public Sub sort_Nchisla_SmalltoLarge()
    Dim A As Variant
    Dim N, p As Integer
    'първи вариант за определяне на броя на въведените елементи
    'ти чрез инструкция за определяне на броя на редовете за
    'област от клетки с начална клетка A1
    N = Sh_sort_N.Range("A1").CurrentRegion.Rows.Count - 1
    'втори вариант за определяне на броя на въведените елементи
    'ти чрез въвеждане на формула в клетка D3 със средствата
    'на VBA
    Sh_sort_N.Range("D3").Formula = "=COUNT(A:A)"
    ReDim A(0 To N - 1) As Integer
    For i = 0 To N - 1
        A(i) = Sh_sort_N.Cells(2 + i, 1)
    Next i
    'алгоритъм за подреждане по метода на мехурчето
    For i = 1 To N - 1
        For j = N - 1 To i Step -1
            If A(j) < A(j - 1) Then
                p = A(j)
                A(j) = A(j - 1)
                A(j - 1) = p
            End If
        Next j
    Next i
End Sub
```

```
End If
Next j
Next i
For i = 0 To N - 1
    Sh_sort_N.Cells(2 + i, 1).Value = A(i)
Next i
End Sub
```

За реализиране на функционалността за автоматично попълване на входни данни чрез случайно пораждање на стойности може да се подходи по аналогичен начин както в Задача 3.

3. Заключение

Предложеният подход и учебни задачи дават възможност учениците да развият и задълбочат своите практически и приложни умения в областта на компетентно използване на информационните технологии и да развият своите алгоритмични знания и умения, като ги приложат в нов контекст и среда за програмиране чрез използване на вградените средства за програмиране на потребителските продукти с общо предназначение.

Работата търпи развитие в следните аспекти:

- създаване на набор от задачи за самостоятелна работа, свързани с творческо прилагане на разгледаните алгоритми;
- реализация на други алгоритми, свързани с тематиката;
- реализация на представените в работата алгоритми чрез използване на вградените средства за програмиране на други потребителски продукти с общо предназначение.

Работата ще е полезна за учители, преподаващи компютърно моделиране и информационни технологии в прогимназията.

БЕЛЕЖКИ

1. Министерство на образованието и науката. *Учебни програми по компютърно моделиране и информационни технологии*, <https://mon.bg/obshto-obrazovanie/uchebni-planove-i-programi-2/uchebni-planove-i-programi-po-klasove/uchebni-programi-po-kompyutarno-modelirane-i-informatsionni-tehnologii>

ЛИТЕРАТУРА

- АНЕВА, С., ТОДОРОВА, Е., 2021. Възможности за развитие на алгоритмични умения на учениците в обучението по предмета „Компютърно моделиране и информационни технологии“ в прогимназията, *Юбилейна международна конференция „Компютърни технологии и приложения“*, стр. 37 – 46.
- ГЪРОВ, К., 2010. Задачите в обучението по информатика и информационни технологии, *Национална конференция „Образованието в информационното общество“*, стр. 95 – 101.
- ДУРЕВА, Д., 2003. *Проблеми от методиката на обучение по информатика и информационни технологии*, Университетско издателство ЮЗУ „Неофит Рилски“, Благоевград.
- КОЖУХАРОВА, Д., 2020. *От дигитална компетентност към дигитална креативност*, Университетско издателство „Тракийски университет“, Стара Загора.
- МАНЕВ, К., МАНЕВА, Н., ХРИСТОВА, В., 2017. *Информатика 8. клас общообразователна подготовка*, Изкуства, София.
- ПАПАНЧЕВА, Р., КАРАДИМИТРОВА, Р., ГЪРОВ, К., 2018. Формиране на дигитални умения в начална училищна възраст – нормативна база и практика в образователната система в България, *Образование и технологии*, бр. 9, стр. 107 – 112.
- ТОДОРОВА, Е., АНЕВА, С., ЧИЛИКОВА, С., ДЕЛЧЕВА, П., 2021. Формиране и развитие на познавателни умения в обучението по „Компютърно моделиране и информационни технологии“ в прогимназията, *Юбилейна международна конференция „Компютърни технологии и приложения“*, стр. 103 – 113.

REFERENCES

- ANEVA, S., TODOROVA, E., 2021. Prospects for Developing Students' Algorithm Skills in Teaching the Subject "Computer Modeling and Information Technologies" in Middle School, *Anniversary International Scientific Conference "Computer Technologies and Applications"*, pp. 37 – 46 (in Bulgarian).
- GAROV, K., 2010. Problems in Informatics and Information Technology Education. *National Conference "Education in the Information Society"*. , pp. 95 – 101 (in Bulgarian).
- DUREVA, D., 2003. *Problems of the methodology of training in informatics and information technologies*. Neofit Rilski, Blagoevgrad (in Bulgarian).

- KOZHUHAROVA, D., 2020. *From digital competence to digital creativity*. Trakiyski universitet, Stara Zagora (in Bulgarian).
- MANEV, K., MANEVA, N., HRISTOVA, V., 2017. *Informatics (text-book), general education, Grade 8*, Izkustva, Sofia (in Bulgarian).
- PAPANCHEVA, R., KARADIMITROVA, R., GAROV, K., 2018. Building Digital Skills at Primary School – Normative Regulation and Pedagogical Practice in Bulgaria, *Education and Technologies*, vol. 9, pp. 107 – 112 (in Bulgarian).
- TODOROVA, E., ANEVA, S., CHILIKOVA, S., DELCHEVA, P., 2021. Forming and Developing Cognitive Skills in Teaching “Computer Modeling and Information Technologies” in Middle School, *Anniversary International Scientific Conference “Computer Technologies and Applications”*, pp. 103 – 113. (in Bulgarian)
- SLAVOVA L., GAROV, K., 2019. Increasing the Digital Competences of Students, *Mathematics and Informatics*, Vol. 62, no 1, pp. 43 – 52.

DEVELOPMENT OF DIGITAL AND ALGORITHMIC SKILLS FOR STUDENTS USING VISUAL BASIC FOR APPLICATIONS

Abstract. This article presents an approach for integrated inter-subject collaboration and creative use of already present digital skills of students for processing data in spreadsheets as well as their programming skills for realizing familiar algorithms from the 5th grade Computer Modeling and Information Technologies school curriculum in new context and programming environment. Six problems are presented along with their solutions, realized with the aid of the integrated programming tools in Excel. Those problems are aimed to the use of linear, branched and cyclic algorithms for automating the realization of the following activities – switching the values of two cells, finding the minimum and maximum value of three or more cells in a worksheet and ordering three or more elements by magnitude.

Keywords: computer modeling; information technologies; digital skills; digital creativity; algorithmic skills; algorithms; teaching methodology; spreadsheets; Visual Basic for Applications

✉ **Dr. Stefka Aneva, Assoc. Prof.**

ORCID iD: 0000-0002-0552-3412

Faculty of Mathematics and Informatics

University of Plovdiv „Paisii Hilendarski“

Plovdiv, Bulgaria

E-mail: stfaneva@uni-plovdiv.bg

✉ **Dr. Elena Todorova, Assist. Prof.**

ORCID iD: 0000-0002-1375-4106

Faculty of Mathematics and Informatics

University of Plovdiv „Paisii Hilendarski“

Plovdiv, Bulgaria

E-mail: etodorova@uni-plovdiv.bg