

## Excel 2013 – Visual Basic for Application

### Instrukcja nr 5

#### Rozszerzenie wiadomości o instrukcji warunkowej i pętlach w języku VBA. Generator liczb pseudolosowych. Tablice. Obsługa błędów.

#### Przykład 1

Uruchom program Microsoft Excel. Zapisz plik pod nazwą **instrukcja5.xlsm** (jako *Skoroszyt programu Excel z obsługą makr*). Przejdź do edytora Visual Basic. W nowym module opracuj program (zapisany w postaci procedury) który będzie wpisywał do komórek **A1:J10** **losowe liczby całkowite** z zakresu od 1 do 10. Jeżeli wartość wprowadzona do danej komórki będzie **większa od 5**, ma zostać ustawiony **jasnozielony kolor tła** tej komórki, w przeciwnym wypadku (wartość mniejsza bądź równa 5) czerwony kolor tła. Przykładowe rozwiązania przedstawiono na rysunkach 1 (sposób I – **For Each**, procedura o nazwie **Losuj\_Tlo**) i 2 (sposób II – dwie pętle **For**, procedura o nazwie **Losuj\_Tlo2**). Rysunek 3 obrazuje przykładowy rezultat działania procedury.

|                                                                                                                                                                                                                                                                                                         |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre>Option Explicit Sub Losuj_Tlo() Dim komorka As Range Range("A1:J10").Clear For Each komorka In Range("A1:J10") Randomize komorka.Value = 1 + Round(Rnd * 9, 0) If komorka.Value &gt; 5 Then komorka.Interior.ColorIndex = 4 Else komorka.Interior.Color = RGB(255, 0, 0) End If Next End Sub</pre> | <p>Definicja zmiennej <b>komorka</b> jako obiektu typu <b>Range</b> (tu reprezentuje komórkę).</p> <p>Dla każdej komórki z zakresu <b>A1:J10</b>. Wartość zmiennej <b>komorka</b> z każdym kolejnym przejściem pętli przyjmuje kolejną dozwoloną wartość z podanego zakresu.</p> <p>Generator liczb pseudolosowych uruchamia słowo kluczowe <b>Randomize</b>, a liczba z przedziału <math>&lt;0, 1</math> jest generowana funkcją <b>Rnd()</b>. Losowanie liczby całkowitej z przedziału od <b>a</b> do <b>b</b>: <math>a + \text{Round}(\text{Rnd} * (b-a), 0)</math></p> <p>Zmiana koloru tła komórki (2 nieco odmienne sposoby).</p> |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Rys. 1. Przykładowe rozwiązanie przykładu 1 (sposób I).

|                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                          |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre>Sub Losuj_Tlo2() Dim i As Integer, j As Integer Range("A1:J10").Clear For i = 1 To 10 For j = 1 To 10 Randomize Cells(i, j) = 1 + Round(Rnd * 9, 0) If Cells(i, j).Value &gt; 5 Then Cells(i, j).Interior.ColorIndex = 4 Else Cells(i, j).Interior.Color = RGB(255, 0, 0) End If Next j Next i Next End Sub</pre> | <p>Wykorzystanie dwóch, zagnieżdżonych pętli <b>For</b> zamiast <b>For Each</b>. Operowanie na komórkach poprzez podanie numeru wiersza i kolumny oraz użycie metody <b>Cells()</b>.</p> |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Rys. 2. Przykładowe rozwiązanie przykładu 1 (sposób II).

|    | A | B | C  | D | E | F  | G | H  | I | J | K |
|----|---|---|----|---|---|----|---|----|---|---|---|
| 1  | 3 | 3 | 10 | 4 | 2 | 10 | 3 | 4  | 1 | 1 |   |
| 2  | 8 | 6 | 2  | 5 | 8 | 1  | 7 | 10 | 7 | 8 |   |
| 3  | 4 | 2 | 1  | 9 | 3 | 2  | 6 | 10 | 5 | 6 |   |
| 4  | 5 | 8 | 8  | 2 | 5 | 6  | 7 | 9  | 8 | 4 |   |
| 5  | 7 | 1 | 5  | 8 | 7 | 5  | 2 | 3  | 3 | 3 |   |
| 6  | 4 | 5 | 4  | 5 | 1 | 4  | 4 | 6  | 7 | 7 |   |
| 7  | 6 | 1 | 4  | 3 | 9 | 6  | 6 | 10 | 6 | 6 |   |
| 8  | 8 | 1 | 5  | 7 | 5 | 2  | 7 | 2  | 2 | 7 |   |
| 9  | 4 | 7 | 4  | 5 | 9 | 5  | 7 | 6  | 9 | 8 |   |
| 10 | 3 | 7 | 2  | 9 | 9 | 2  | 5 | 7  | 2 | 9 |   |
| 11 |   |   |    |   |   |    |   |    |   |   |   |

Rys. 3. Przykładowy rezultat działania programu LosujTlo oraz LosujTlo2.

### Przykład 2

Opracuj program (zapisany w postaci procedury) o nazwie **Sumuj\_dodatnie**, który będzie obliczał **sumę liczb dodatnich** podawanych przez użytkownika w oknie(ach) **InputBox()**. Po wprowadzeniu liczby niedodatniej (która nie jest już uwzględniana w dalszych obliczeniach) ma on kończyć swoje działanie i zapytać użytkownika (okno **InputBox()**) o adres komórki, do której ma zostać wstawiony wynik. W celu przechwycenia możliwych błędów (np. wprowadzanie niepoprawnych wartości) wykorzystaj polecenie **On Error GoTo** i zwróć odpowiedni komunikat **MsgBox()**. Przykładowe rozwiązanie przedstawiono na rysunku 4.

```

Sub Sumuj_dodatnie()
Dim a As Single, suma As Single
Dim adres As String
suma = 0
On Error GoTo Blad
a = InputBox("Sumowanie liczb dodatnich", "Podaj liczbę", "Tu wpisz wartość")
While a > 0
    suma = suma + a
    a = InputBox("Sumowanie liczb dodatnich", "Podaj liczbę", "Tu wpisz wartość")
Wend
adres = InputBox("Podaj adres komórki, w której zostanie zwrócony wynik", _
"Podaj adres komórki", "Tu wpisz adres komórki (np. A10)")
Range(adres).Value = suma
Exit Sub
Blad:
MsgBox ("Podano niepoprawne dane!" & vbCrLf & "Opis błędu w języku angielskim: " & _
& Err.Description)
Err.Clear
End Sub

```

*Instrukcja **On Error GoTo Nazwa\_etykiety** służy do obsługi błędów. Zazwyczaj jest umieszczana na początku programu. W przypadku wykrycia błędu w którejkolwiek linii kodu, następuje przeskoczenie do fragmentu kodu znajdującego się pod podaną nazwą etykiety (tu: **Blad**).*

***Exit Sub** – instrukcja kończąca pracę procedury. Bez niej kod obsługi błędu wykona się nawet, jeżeli błąd nie nastąpi.*

*Obiekt **Err** zawiera szczegóły błędu.*

Rys. 4. Przykładowe rozwiązanie przykładu 2.

### Przykład 3

Zapoznaj się z kodem programu **Tablice** przedstawionym na rysunku 5, który dotyczy obsługi tablic w języku VBA, a następnie uruchom go w środowisku MS Excel i sprawdź jego działanie.

Zadaniem programu jest wypełnienie **10-elementowej tablicy** o nazwie **tablica1** (posiadającej numery indeksów od 1 do 10) liczbami z zakresu od 2 do 20 z krokiem 2. Następnie zawartość tablicy jest drukowana w 1 kolumnie arkusza programu Excel, rozpoczynając od 15 wiersza.

```

Sub Tablice()
Dim tablica1(1 To 10) As Integer
Dim i As Integer, j As Integer
Dim k As Variant
j = 1
For i = 2 To 20 Step 2
    tablica1(j) = i
    j = j + 1
Next
j = 0
For Each k In tablica1
    Cells(15 + j, 1).Value = k
    j = j + 1
Next
End Sub

```

Deklaracja jednowymiarowej, 10-elementowej tablicy o nazwie **tablica1**, przechowującej dane typu *Integer*. Posiada ona numery indeksów od 1 do 10.

Rys. 5. Obsługa tablic.

### Przykład 4

Opracuj program (zapisany w postaci procedury) o nazwie **Losowanie**, który będzie losował **6 liczb całkowitych** z zakresu od **1** do **49** bez zwracania – **losowanie lotto**. Użytkownik wpisuje **6** typowanych liczb do komórek **D10:I10** – należy sprawdzić kompletność i poprawność podanych danych, w tym, czy wprowadzona liczba jest całkowita i należy do przedziału <1, 49>. Zakłada się jednak, że tak jak przy wypełniania kuponu, podane liczby nie będą się powtarzały (nie trzeba tego sprawdzać w aplikacji). Wyniki losowania są umieszczane w komórkach **B3:B8**. Następnie należy rozszerzyć funkcjonalność programu według poleceń zawartych w zadaniu 4.

Opis zasady działania programu:

- Definicja tablicy **liczby** (rozmiar tablicy: **49**; numery indeksów: od **1** do **49**; przechowywane wartości: typ **Integer**) – będzie ona przechowywała wszystkie możliwe wartości liczbowe.
- Definicja tablicy **wylosowane\_liczby** (rozmiar tablicy: **6**; numery indeksów: od **1** do **6**; przechowywane wartości: typ **Integer**) – będzie ona przechowywała wylosowane liczby.
- Sprawdzenie poprawności zawartości komórek **D10:I10** – dane te wprowadza każdorazowo użytkownik.
- n=49**.
- Wypełnienie tablicy **liczby** wartościami liczbowymi z zakresu od **1** do **n** z krokiem **1**.
- Sześciokrotne wykonanie kroków (**i=1 To 6**):
  - Losowanie indeksu (losowej liczby) z zakresu od **1** do **n** i przypisanie go do zmiennej **wylosowanyIndeks**.
  - Umieszczenie wartości **liczby(wylosowanyIndeks)** na **i-tej** pozycji tablicy **wylosowane\_liczby**.
  - Umieszczenie w tablicy **liczby** na pozycji **wylosowanyIndeks** wartości znajdującej się na pozycji o indeksie **n** (przypisanie ostatniej wartości przechowywanej na liście **liczby**).
  - n=n-1**.
  - Wpisanie wartości wylosowanej liczby w odpowiednim wierszu (rozpoczynając od **3**) w kolumnie **A**.
- Miejsce na uzupełnienie o fragment kodu (w miejscu komentarza), który będzie odpowiadał za sprawdzenie wyników, według poleceń zadania 4.

Sugerowany wygląd arkusza wraz z przykładowym rezultatem działania programu (już po uwzględnieniu rozwiązania zadania 4) przedstawiono na rysunku 6. Proponowany kod procedury **Losowanie** zamieszczono na rysunku 7 – jest to tylko jedna z wielu możliwości rozwiązania przedstawionego problemu.

|    | A | B                        | C | D                       | E  | F  | G | H | I  |
|----|---|--------------------------|---|-------------------------|----|----|---|---|----|
| 1  |   |                          |   |                         |    |    |   |   |    |
| 2  |   | <b>Wyniki losowania:</b> |   |                         |    |    |   |   |    |
| 3  |   | 14                       |   |                         |    |    |   |   |    |
| 4  |   | 30                       |   |                         |    |    |   |   |    |
| 5  |   | 42                       |   |                         |    |    |   |   |    |
| 6  |   | 5                        |   |                         |    |    |   |   |    |
| 7  |   | 22                       |   |                         |    |    |   |   |    |
| 8  |   | 33                       |   |                         |    |    |   |   |    |
| 9  |   |                          |   | <b>Typowane liczby:</b> |    |    |   |   |    |
| 10 |   |                          |   | 22                      | 16 | 10 | 6 | 3 | 33 |
| 11 |   |                          |   |                         |    |    |   |   |    |
| 12 |   | <b>Trafionych:</b>       | 2 |                         |    |    |   |   |    |

Rys. 6. Sugerowany wygląd arkusza wraz z przykładowym rezultatem działania programu.

```

Sub Losowanie()
Dim liczby(1 To 49) As Integer
Dim wylosowane_liczby(1 To 6) As Integer
Dim i As Integer, n As Integer, wylosowanyIndeks As Integer
Dim komorka As Range
For Each komorka In Range("D10:I10")
    If Not IsNumeric(komorka) Or IsEmpty(komorka) Then
        Range("B3:B8, C12").Value = ""
        MsgBox ("Podano błędne dane!")
        Exit Sub
    Else
        If komorka <> Int(komorka) Or komorka.Value < 1 Or komorka.Value > 49 Then
            Range("B3:B8, C12").Value = ""
            MsgBox ("Podano błędne dane!")
            Exit Sub
        End If
    End If
Next
n = 49
For i = 1 To n
    liczby(i) = i
Next
For i = 1 To 6
    Randomize
    wylosowanyIndeks = 1 + Round(Rnd * (n - 1), 0)
    wylosowane_liczby(i) = liczby(wylosowanyIndeks)
    liczby(wylosowanyIndeks) = liczby(n)
    n = n - 1
    Cells(i + 2, 2).Value = wylosowane_liczby(i)
Next
'Sprawdzenie ilości poprawnie wytypowanych liczb
End Sub

```

Funkcja **Int** zwraca część całkowitą liczby.

Rys. 7. Proponowany kod programu Losowanie Lotto.

### Zadania do samodzielnego rozwiązania

Wszystkie poniższe zadania wykonaj w pliku **instrukcja5.xlsm** – jeden skoroszyt może zawierać wiele makr. Każdorazowo sprawdzaj poprawność działania utworzonego makra.

1. Zmodyfikuj kod programu **Losuj\_tlo** w taki sposób, by losował liczby całkowite z zakresu od 2 do 15 i w zależności od wartości wylosowanej liczby zmieniał kolor tła danej komórki według poniższego schematu:

dla liczb z zakresu  $\langle 2, 5 \rangle$  – **jasnoniebieski**,  
 dla liczb z zakresu  $\langle 6, 10 \rangle$  – **pomarańczowy**,  
 dla liczb z zakresu  $\langle 11, 15 \rangle$  – **szary**.

- Zmodyfikuj kod programu otrzymanego w **zadaniu 1** w taki sposób, by losowane były liczby rzeczywiste z podanego zakresu.
- Przekopiuj kod procedury **Sumuj\_dodatnie** i nadaj utworzonej kopii nazwę **Srednia\_z\_dodatnich**. Zmodyfikuj następnie jej kod w taki sposób, by liczyła **średnią z liczb dodatnich wprowadzanych przez użytkownika**. Dodaj ponadto czytelny komunikat **MsgBox()** w przypadku, gdy nie podano żadnej liczby dodatniej.
- Przekopiuj kod procedury **Tablice** i nadaj utworzonej kopii nazwę **Tablice2**. Zmodyfikuj kod programu w taki sposób, by zawierał on definicję **21-elementowej tablicy** wypełnionej liczbami **rzeczywistymi** z zakresu od **10** do **20** z krokiem **0.5**.
- Uzupełnij program **Losowanie** o fragment kodu (w miejscu komentarza), który będzie odpowiadał za sprawdzenie wyników, tzn. obliczenie, ile z liczb podanych przez użytkownika zostało wylosowanych. Wynik należy zamieścić w komórce **C12**.
- Opracuj program (zapisany w postaci procedury) o nazwie **Pierwiastki**, który ma obliczać wartości pierwiastków **3 stopnia** dla liczb dodatnich umieszczonych w komórkach **A1:A10** (należy sprawdzić poprawność podanych danych) i zwracać wyniki w odpowiadającym im wierszach w kolumnie **B**, zaokrąglone w przypadku liczby rzeczywistej do **3 miejsc po przecinku**. Jeżeli w którejś komórce wartość jest mniejsza bądź równa **0**, żadne obliczenie nie jest wykonywane. Przykładowy rezultat działania programu przedstawiono na rysunku 8.

|    | A    | B     |
|----|------|-------|
| 1  | 8    | 2     |
| 2  | 0    |       |
| 3  | 27   | 3     |
| 4  | -8   |       |
| 5  | 64   | 4     |
| 6  | -100 |       |
| 7  | 10   | 2,154 |
| 8  | 6    | 1,817 |
| 9  | -1   |       |
| 10 | 100  | 4,642 |

Rys. 8. Przykładowy rezultat działa programu Pierwiastki.

- Opracuj program (zapisany w postaci procedury) o nazwie **Pesel**, który na podstawie **10** pierwszych cyfr numeru **PESEL** (podanych w komórkach **B3: K3**) ma obliczać **jedenastą**, kontrolną cyfrę tego numeru i zwrócić wynik do komórki **L3**. Zabezpiecz program przed podaniem błędnych danych. Przykładowy rezultat działania programu przedstawiono na rysunku 9.

|   | A | B | C | D | E | F | G | H | I | J | K | L |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 |   |   |   |   |   |   |   |   |   |   |   |   |
| 2 |   |   |   |   |   |   |   |   |   |   |   |   |
| 3 |   | 2 | 5 | 0 | 2 | 2 | 2 | 8 | 9 | 1 | 1 | 6 |

Rys. 9. Przykładowy rezultat działa programu Pesel.