



VBA voor doe het Zelfers deel 22

Handleiding van Helpmij.nl

Auteur: leofact

december 2015

“ Dé grootste en gratis computerhelpdesk van Nederland ”

Vorige aflevering

In de vorige aflevering werden de regular expressions behandeld. Voor VBA zijn deze beschikbaar in de Microsoft VBScript-bibliotheek. Hiermee is het mogelijk om op eenvoudige wijze invoermaskers te maken waarmee invoer op het juiste format is te controleren. Naast uitleg van het gebruik van de methoden, eigenschappen en speciale tekens uit deze bibliotheek werden een paar voorbeelden uitgewerkt. Hieronder een postcode, een e-mailadres, een BSN en een Nederlandse IBAN. Bij de laatste twee werd naast de controle op het format ook een rekenkundige controle gegeven om de juistheid van het nummer te kunnen verifiëren. Naast de invoer met een inputbox werd ook behandeld hoe je de functie in een werkblad kan opnemen. De procedures waren uitgewerkt in de [bijlage](#).

In deze aflevering

Dit keer wordt een lezersvraag behandeld. Deze kwam binnen op de redactie e-mail. Dit als reactie op de oproep om onderwerpen voor de VBA-reeks aan te dragen. Arie Alblas mailde het volgende:

“Allereerst mijn complimenten voor de site van Helpmij.nl. Graag zal ik in het onderwerp VBA lessen, het Random sorteren en nummers in willekeurige volgorde presenteren, behandeld willen zien. Dit voor gebruik in Excel.”

Mij was nog niet duidelijk wat er precies bedoeld werd en er volgde nog wat e-mailverkeer waaruit het volgende naar voren kwam:

Ik bedoel het in willekeurige volgorde plaatsen van namen in 5 kolommen, welke wordt gepresenteerd waarbij men maar 1 keer tegen een tegenstander mag spelen. Men mag dus niet meer dezelfde persoon tegen komen. Het betreft de sport Jeu de Boules (Pétanque). Door middel van loting wil ik een lijst presenteren in Excel, waarop men kan zien tegen welke speler men moet spelen. Dit gaat over 5 rondes waarin men niet 2 maal tegen dezelfde speler speelt (dus elke ronde een andere tegenstander). Het maximaal aantal spelers wat opgegeven kan worden is 124. Dit kan variëren hoor. De spelers spelen 1 tegen 1 en met z'n allen tegelijk. Iedere speler krijgt bij inschrijving een nummer, dit is de notatie volgorde in de kolom. Nu moet er een loting plaatsvinden, waar in elke kolom (ronde) een ander nummer (speler) wordt genoteerd. Dit graag met een random-formule. Ik zal een voorbeeldbestandje meesturen. Hieronder zie je een screenshot van het voorbeeldbestand:

Naam	Ronde 1	Ronde 2	Ronde 3	Ronde 4	Ronde 5
Jan					
Piet					
Klaas					
Kees					
Henk					
Wil					
Janny					
Karel					
Gerrit					
Siem					
Leen					
Joost					
Freek					
Rene					
Robbert					
Nijs					
Fop					
Douwe					
Annemiek					
Ivar					

Een jeu de boules-toernooi dus met een maximum van 124 spelers die in vijf verschillende rondes nooit tegen dezelfde speler spelen.

Vroeger, op vakantie met het gezin, werd er wel jeu de boules gespeeld. Ik weet er echter nog weinig van en al helemaal niets over toernooi-regels. Dus maar wat rond gegoogled. Het wordt wel een round robin-toernooi genoemd. Al snel blijkt dat er voor bovenstaande opdracht genoeg sites zijn waarin je een heel toernooi kan organiseren. Om er een paar te geven:

http://challonge.com/tournament/bracket_generator

<http://www.printyourbrackets.com/generator.php>

En een heel complete, waarbij je voor veel soorten sport verschillende vormen van een toernooi kan opzetten: <http://www.konkuri.com/>.

Eigenlijk is het nu niet nodig om verder te werken; het toernooi kan gegenereerd worden. Echter, niet in Excel en dat was de vraag. Bovendien zit er ook een interessante uitdaging in. Een en ander is ook dit keer uitgewerkt in de [bijlage](#).

Random

De vraag was dus een willekeurig schema. Daar zijn willekeurige getallen voor nodig. Dat lijkt niet zo'n probleem. Toch is het dat wel voor een computer, die heeft uiteindelijk geen wil en kan alleen rekenen (dat geldt althans voor de computers waar wij thuis ons officepakketje op draaien). Zelfs als we zelf een willekeurig getal opschrijven blijkt dit niet bepaald echt willekeurig te zijn. Dat komt omdat we geneigd zijn om voorwaarden aan de willekeur te stellen. Je zult bijvoorbeeld niet snel 4 gelijke getallen opschrijven, omdat dit niet willekeurig *lijkt*. In werkelijkheid kan dat juist wél voorkomen bij een echt willekeurige reeks. Een kenmerk van willekeur is per slot van rekening dat het niet aan voorwaarden voldoet. Om een computer toch zo ver te krijgen dat er met een min of meer willekeurig getal kan worden gewerkt zijn er diverse pseudo-toevalsgenerators ontwikkeld. Het zijn algoritmes die een moeilijk voorspelbaar getal als uitkomst hebben. Wanneer je daarvoor interesse hebt, kun je er meer over lezen in [deze link](#). In VBA hoeven we ons niet druk te maken over algoritmes. Er is namelijk voorzien in de functie RND. Deze functie geeft een random Single-variabele tussen nul en één als resultaat. Het gaat om pseudo-toevalsgetallen, herhaling van de reeks is daarom mogelijk. De functie Randomize stelt de generator steeds opnieuw in aan de hand van een toegevoegd getal. Dit getal wordt "seed" (zaad) genoemd. Wanneer er geen getal wordt ingesteld, wordt het serieel-datum en tijd getal genomen. Dat is daardoor automatisch steeds een ander getal. Een simpele procedure die hier gebruik van maakt kan er als volgt uitzien:

```

(Algemeen) | WillekeurigGetal
Sub WillekeurigGetal ()
'-----
' Procedure : WillekeurigGetal
' Date      : 28-10-2015
' Purpose   : genereer 10000 random getallen
'-----
Dim y As Integer
'activeer juiste blad
Blad3.Activate
'zet automatisch berekenen uit (versnelt procedure)
Application.Calculation = xlCalculationManual
'doorloop lus voor 10000 willekeurige getallen
Do
    y = y + 1
    'stel de generator in
        Randomize
    Cells(y + 2, 1) = Rnd
Loop While y < 10000
'berekenen weer aan
Application.Calculation = xlCalculationAutomatic
End Sub
    
```

Vaak gebruik je echter een bepaalde range getallen. Deze range kun je als volgt bepalen:

$$\text{Uitkomst} = (i\text{Hoogste} + 1 - i\text{Laagste}) * \text{Rnd} + i\text{Laagste}$$

Wanneer je alleen gehele getallen wilt, gebruik je de integer functie *Int*.
*Uitkomst = Int((iHoogste + 1 - iLaagste) * Rnd + iLaagste)*

Hiermee wordt de uitkomst naar beneden afgerond tot een geheel getal. Een voorbeeld zie je hieronder:

```

(Algemeen) | WillekeurigBegrensd
Sub WillekeurigBegrensd()
'-----
' Procedure : WillekeurigBegrensd
' Date      : 28-10-2015
' Purpose   : genereer 10000 begrensde random getallen
'-----
Dim y As Integer
Dim iLaagste As Integer
Dim iHoogste As Integer
'activeer juiste blad
Blad3.Activate
'zet automatisch berekenen uit (versnelt procedure)
Application.Calculation = xlCalculationManual
'bepaal de grenzen
iLaagste = 1
iHoogste = 10
'doorloop lus voor 10000 begrensde willekeurige getallen
Do
    y = y + 1
    'stel de generator in
    Randomize
    Cells(y + 2, 1) = Int((iHoogste + 1 - iLaagste) * Rnd + iLaagste)
Loop While y < 10000
'berekenen weer aan
Application.Calculation = xlCalculationAutomatic
End Sub
    
```

Toernooi

Gewapend met deze kennis lijkt het nu simpel om het toernooi op te zetten. Bijvoorbeeld op de volgende manier:

1. Zet vijf rondes op met een For Next-lus.
2. Roep een willekeurig getal op voor speler 1.
3. Start een Do-lus.
4. Kijk of deze speler al is toegewezen.
5. Zo ja, ga terug in de lus, zo nee verlaat de lus en ga verder.
6. Start nog een Do-lus.
7. Roep een willekeurig getal op voor speler 2.
8. Heeft deze speler al eerder tegen speler 1 gespeeld? Ga dan terug in de lus, zo niet; verlaat de lus en ga verder.
9. Zet de uitkomsten in het schema.
10. Ga hiermee door tot alle spelers een tegenstander hebben.
11. Doe dit voor de vijf rondes (Next) en klaar.

Dit kan op de volgende manier worden uitgewerkt (hierbij wordt een vrije ronde toegevoegd als er een oneven aantal spelers is opgegeven):

```

(Algemeen) | JdBTournoi
'-----
' Procedure : JdBTournoi
' Date      : 28-10-2015
' Purpose   : Genereer een willekeurig speelschema
' Let op! Kan Excel laten crashen bij te weinig spelers
'-----

Dim iLaatste As Integer
Dim iRonde As Integer
Dim iSpeler1 As Long
Dim iSpeler2 As Long
Dim y As Integer
Dim x As Integer
'activeer juiste blad
Blad2.Activate
'reset schema; verwijderd eventuele lege cellen
[B3:B125].Sort [b3]
[C3:K125].ClearContents
'stel aantal rondes en spelers in.
iRonde = 5
iLaatste = Range("B3").End(xlDown).Row - 2
'indien oneven, voeg vrije ronde toe
If iLaatste Mod 2 > 0 Then
    iLaatste = iLaatste + 1
    Cells(iLaatste + 2, 2) = "vrije ronde"
End If
'doorloop het aantal rondes
For x = 1 To iRonde
    'zoek eerste speler willeurig
    Do
        iSpeler1 = Willekeurig(iLaatste)
        If Cells(iSpeler1 + 2, x + 2) = "" Then
            'zoek willekeurige tweede speler
            Do
                iSpeler2 = Willekeurig(iLaatste)
                'niet eerder tegen elkaar?
                If Cells(iSpeler2 + 2, x + 2) = "" And _
                    iSpeler1 <> iSpeler2 Then
                    If Application.CountIf(Range(Cells(iSpeler1 + 1, 3), _
                        Cells(iSpeler1 + 1, 8)), iSpeler2) = 0 Then
                        'zet de spelers in het schema
                        Cells(iSpeler1 + 2, x + 2) = iSpeler2
                        Cells(iSpeler2 + 2, x + 2) = iSpeler1
                        Exit Do
                    End If
                End If
            Loop
        End If
    Loop
    ' ga door tot er geen lege cellen meer zijn in de kolom.
    Loop While Application.WorksheetFunction.CountBlank _
        (Range(Cells(2, x + 2), Cells(iLaatste + 1, x + 2))) > 0
'volgende ronde
Next
Application.EnableEvents = True
End Sub

```

Voor het eigenlijke genereren van een willekeurig getal wordt de volgende functie gebruikt:

```

(Algemeen) | Willekeurig
Function Willekeurig(ByVal iLaatste As Integer) As Integer
'geef een willekeurig getal tussen 1 en iLaatste als resultaat
Randomize
Willekeurig = Int((iLaatste) * Rnd + 1)
End Function

```

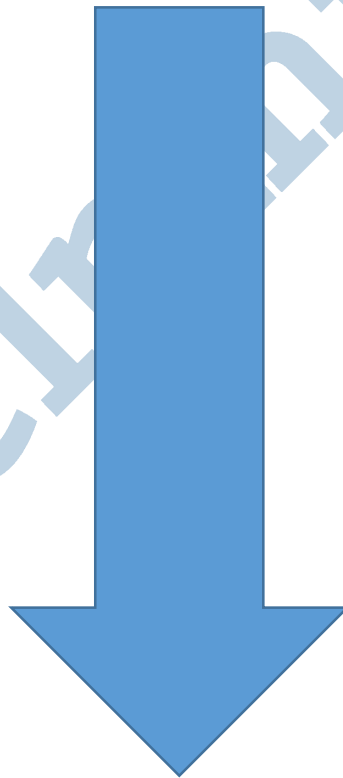
Wanneer er veel spelers mee doen is er geen vuiltje aan de lucht. Het schema staat in no-time klaar. Mooi, probleem opgelost dus. Iemand met een beetje inzicht in wiskunde heeft echter al in de gaten dat er een probleem gaat ontstaan wanneer er weinig spelers mee doen. Je kunt dan al snel in de situatie terecht komen dat er geen unieke spelers-paren meer zijn te genereren, terwijl de ronde nog niet gevuld is. Dat merk je bij bovenstaande routine; Excel crasht dan namelijk. Dat kan natuurlijk niet de bedoeling zijn. Er zijn daarvoor mogelijk trucjes te bedenken; bijvoorbeeld opnieuw beginnen, tot er bij toeval een juist schema is uitgerekend. Dat komt op mij echter niet echt netjes over. Uitrekenen dan maar?

Round-robin

Daar kwam ik echter niet zo één twee drie uit. Inderdaad een gebrekkig wiskundig-inzicht. Gelukkig is er dan Google. Round-robin blijkt een gecombineerd optellend en aftellend schema met een twist; namelijk een gefixeerd cijfer. Dit blijkt onder meer in netwerkverkeer te worden gebruikt om de belasting te verdelen. Dit schema is overigens niet bedacht door ene Robin. De naam is een verbastering van ribbon; het verschuivende lint wat als het ware wordt gebruikt. De volgende [Wiki](#) (Engels) geeft duidelijk het algoritme weer met behulp van een paar verschillende schema's (er zijn meerdere vormen). Met een beetje gepuzzel blijkt dat redelijk eenvoudig in een procedure te vangen. Het is nog maar een kleine stap om de spelersnummers te vervangen door de namen van de spelers. Daarbij kan vooraf kan een keuze worden gemaakt voor het aantal rondes door in het werkblad naast "aantal rondes" het gewenste aantal te kiezen. Bij de start van de routine krijgen de spelers een willekeurig nummer toegewezen en worden in die volgorde gesorteerd. Aan het eind worden de spelers dan weer op de oorspronkelijke volgorde (inschrijfnummer) gesorteerd. Zo wordt het in het berekende schema onvoorspelbaar welke spelers tegen wie gaan spelen.

De routine kan als volgt geprogrammeerd worden:

Ga verder




```

(Algemeen) RoundRobin_Tournoi
Sub RoundRobin_Tournoi ()
'-----
' Procedure : RoundRobin_Tournoi
' Date      : 28-10-2015
' Purpose   : Genereer schema met Round Robin algoritme
'-----
Dim iSpelers As Integer
Dim iRonde As Integer
Dim iSp1 As Integer
Dim iSp2 As Integer
Dim y As Integer
Dim x As Integer
Dim rSpelers As Range
Dim lVrij As Long
'activeer juiste blad
Blad1.Activate
'reset schema
Application.ScreenUpdating = False
[D3:DV126].ClearContents
[B3:B126].ClearContents
'maak een Range variable van de spelers
Set rSpelers = Range("C3", Range("C3").End(xlDown))
iSpelers = rSpelers.Count
With Application.WorksheetFunction
'lege speelnummers zijn niet toegestaan
If .CountA([C3:C126]) > iSpelers Then
MsgBox "Vul eerst alle speelnummers" & vbLf _
& "tot het hoogst in gebruik zijnde speelnummer" & vbLf _
& "Er kan geen schema worden gegenereerd"
Exit Sub
End If
'voeg een vrije ronde toe bij een oneven aantal speler.
If iSpelers Mod 2 > 0 Then
Cells(iSpelers + 3, 3) = "vrije ronde"
iSpelers = iSpelers + 1
Set rSpelers = rSpelers.Resize(iSpelers)
End If
'zet de vrije ronde altijd onderaan
If .CountIf(rSpelers, "vrije ronde") > 0 Then
Cells(.Match("vrije ronde", rSpelers, 0) + 2, 2) = iSpelers
lVrij = -1
End If
'maak de volgorde van de spelers willekeurig
For y = 3 To iSpelers + 2
'laat de vrij ronde ongemoeid
If Cells(y, 3) <> "vrije ronde" Then
'start loop voor een willekeurig ongebruikt nummer
Do
Randomize
iSp1 = Int((iSpelers + lVrij) * Rnd + 1)
'uniek nummer?
If .CountIf(Range("B3", "B" & iSpelers + 3), iSp1) = 0 Then
'gebruik het dan
Cells(y, 2) = iSp1
'klaar met de loop
Exit Do
End If
'vervolg op zoek naar niet gebruikt nummer
Loop
End If
'volgende speler
Next y
'sorteer de spelers op de willekeurige volgorde
Range("A3", Cells(iSpelers + 3, iSpelers + 3)).Sort [b2]
'stel het aantal rondes in.
iRonde = iSpelers - 1
'haal de ronde op uit range: "rondes"
If IsNumeric([rondes]) And [rondes].Value < iRonde Then _
iRonde = [rondes].Value
'doorloop het aantal rondens
For x = 4 To iRonde + 3
'fixeer speler 1
iSp1 = 1
'geef speler 2 een waarde van hoog naar laag
iSp2 = iSpelers - (x - 4)
'doorloop de helft van het aantal spelers
For y = 1 To (iSpelers / 2)
'Neem bij de tweede speler vanaf de tweede ronde
'het bovenste getal van de vorige ronde
If y = 2 And x > 4 Then iSp1 = _
.Match(Cells(3, x - 1), rSpelers, 0)
'zoek de naam bij het nummer en plaats in de cel
Cells(iSp1 + 2, x) = .Index(rSpelers, iSp2)
Cells(iSp2 + 2, x) = .Index(rSpelers, iSp1)
'tel speler 2 af tot 2, ga dan naar de hoogste
iSp2 = IIf(iSp2 = 2, iSpelers, iSp2 - 1)
'tel speler 1 op tot de hoogste, ga dan naar 2
iSp1 = IIf(iSp1 = iSpelers, 2, iSp1 + 1)
'volgende speler
Next y
'volgende ronde
Next x
End With
'sorteer alfabetisch, laat de vrije ronde ongemoeid.
Range("A3", Cells(iSpelers + (2 + lVrij), iSpelers + 3)).Sort [A2]
Set rSpelers = Nothing
End Sub

```

Ook hierbij wordt van de functie Willekeurig gebruik gemaakt, op dezelfde manier als bij de vorige routine.

De procedure blijkt te werken zoals Arie bedoeld heeft. Voor nu; einde opdracht. De willekeur is echter nog te vergroten door een gefixeerd-getal te nemen dat ook willekeurig wordt gegenereerd. Daarvoor moet "de twist", ofwel het punt in de reeks waar het gefixeerde getal wordt overgeslagen, anders geprogrammeerd worden. Dat laat ik graag aan de eigen fantasie over.

Er kan een mooie uitdaging worden gezien in een andere oplossing van het algoritme. Ik heb me eruit gered door gebruik te maken van de IIF functie. Een echt algoritme door middel van een berekening zou een veel mooiere oplossing kunnen zijn. Voel je vooral uitgedaagd om dit te proberen. Het wordt zeer gewaardeerd wanneer je deze instuurt naar redactie@helpmij.nl. Goede oplossingen worden gegarandeerd geplaatst.

Hiermee komt deze aflevering tot een einde.

Samenvatting

Deze aflevering is gebaseerd op de vraag van Arie Alblas om een toernooischema te maken waarin spelers in willekeurige volgorde tegen elkaar spelen, zonder de dezelfde tegenstander meer dan één keer te treffen. Hiervoor is de functie RND gebruikt, waarbij het gebruik van deze functie werd uitgelegd. Daarbij werd kort verklaard waaraan een willekeurig getal moet voldoen en waar rekening mee gehouden moet worden bij het programmeren hiervan. Er werden twee toernooischema-generatoren geprogrammeerd. De eerste werd uitsluitend op toeval gebaseerd, maar deze kon vastlopen. De tweede oplossing werd gebaseerd op het round-robin algoritme en geeft altijd een correct schema. Andere oplossingen worden uitdrukkelijk op prijs gesteld, evenals onderwerpen voor een volgende aflevering. Ook deze kun je sturen naar redactie@helpmij.nl.