



## **De bijeenkomst van zaterdag 9 oktober 2021**

**Deze is na lange tijd weer in Bilthoven van 11 tot 15 uur**

**Let op: vanaf 25 september zijn de RIVM maatregelen gewijzigd. Bezoekers moeten hun Corona QR code kunnen overleggen. Dat kan op de smartphone of op papier.**

### **Het programma**

11.00 - Kort verslag van de werkgroepen door Willem Ouwerkerk

11:20 - Rosetta code voorbeelden, door Albert Nijhof

12:30 - Pauze

13:15 - De EmbeddedForth methode, door Willem Ouwerkerk

14:15 - Iedereen kan kort iets vertellen over de elektronica, of projectjes waar hij mee bezig is.  
Daarbij kunnen ook de beamer en het scherm gebruikt worden.

14:45 - Voorbereiding voor contact met Forth2020  
Als de techniek ons niet in de steek laat,  
kunnen we aan het einde van de bijeenkomst contact maken met Forth2020 en mogelijk vragen stellen aan Charles (Chuck) Moore, de uitvinder van Forth.

15.?? - Sluiting

### **Internationale Zoom sessie Forth2020**

Er is een internationale Forth bijeenkomst via Zoom op de tweede zaterdag van elke maand. Een vorige keer was het onderwerp libraries een verhit punt van discussie. Er zijn meerdere presentaties van elk 15 minuten in het Engels. De Zoom sessie wordt gehost door Peter Forth in Brazilië en Ulrich Hoffmann in Duitsland. Opdat de sessie over de hele wereld op een handige tijd gevolgd kan worden, is voor Nederland 16:00 de begintijd. De oude sessies zijn ook nog op youtube te bekijken. Zie ook <https://www.forth2020.org/home>

Om te kijken met zoom onze tijd 4 uur 's middags :

You are heartily invited for this saturday 9 October at 14 UTC to our Forth2020 Zoom meeting.

[zoom.forth2020.org](https://zoom.forth2020.org)

**password : forth2020**

Charles Moore heeft toegezegd al eerder contact met onze Forth-groep te willen maken aan het einde van onze bijeenkomst rond 3 uur 's middags.

## **Rosetta code voorbeelden, door Albert Nijhof**

Op de website 'rosettacode' staat een hele reeks programmeeropdrachten. Het was ooit de bedoeling (het is al een heel oude site), dat iedereen die dat maar wilde, oplossingen zou insturen in allerlei programmeertalen. Een van de opgaven bestaat uit het inventariseren van een DNA-keten.

Die wil ik als uitgangspunt gebruiken bij mijn verhaaltje op de komende forthbijeenkomst.

[https://rosettacode.org/wiki/Bioinformatics/base\\_count](https://rosettacode.org/wiki/Bioinformatics/base_count)

You are encouraged to solve this task according to the task description, using any language you may know.

Given this string representing ordered DNA bases:

```
CGTAAAAAATTACAACGTCCTTTGGCTATCTCTTAAACTCCTGCTAAATG
CTCGTGCTTTCCAATTATGTAAGCGTTCCGAGACGGGGTGGTCGATTCTG
AGGACAAAGGTCAAGATGGAGCGCATCGAACGCAATAAGGATCATTGAT
GGGACGTTTCGTGACAAAGTCTTGTTTCGAGAGTAACGGCTACCGTCTT
CGATTCTGCTTATAAACTATGTTCTTATGAAATGGATGTTCTGAGTTGG
TCAGTCCCAATGTGCGGGGTTTCTTTTAGTACGTCGGGAGTGGTATTATA
TTAATTTTTCTATATAGCGATCTGTATTTAAGCAATTCATTTAGGTTAT
CGCCGCGATGCTCGGTTCCGACCGCCAAGCATCTGGCTCCACTGCTAGTG
TCCTAAATTTGAATGGCAAACACAAATAAGATTTAGCAATTCGTGTAGAC
GACCGGGGACTTGATGATGGGAGCAGCTTTGTTAAACTACGAACGTAAT
```

Task

- "Pretty print" the sequence followed by a summary of the counts of each of the bases: (A, C, G, and T) in the sequence
- print the total count of each base in the string.

## **De EmbeddedForth methode, door Willem Ouwerkerk**

Dit is een attachment bij de mailing:

### **Uitnodiging oktober 2021.pdf**

## **Cursorpositie in VT100, door Willem Jager**

In een applicatie, tijdens het afdrukken van tekst op een terminal, is het vaak handig om te weten hoeveel ruimte er op de regel nog beschikbaar is, om te voorkomen dat de tekst buiten beeld verdwijnt of op de volgende regel verder gaat.

Op zich is het mogelijk om de afgedrukte tekens te tellen, maar dat geeft complicaties met multi-byte tekens en niet-afdrukbare codes.

VT100-terminals hebben hiervoor het "Cursor Position Report" (CPR), dat informatie geeft over de positie van de cursor op het scherm. En terminal emulators die aan de bijbehorende ECMA standaard voldoen hebben die faciliteit ook.

Vanuit de applicatie kan je dit CPR aanvragen door het sturen van de string "\033[6n" (eerste teken is ESC) naar de terminal, vanuit forth met bijvoorbeeld met 27 emit ." [6n"

Als we dit definiëren:

```
: CPR 27 emit ." [6n" ;
```

en dan CPR uitvoeren krijgen we 'ok' en dan op de volgende regel 5R. Eigenlijk verwachten we iets als <esc>[<regel>;<kolom>R dus kennelijk valt er een deel van het resultaat weg, maar er gebeurt wel iets.

Hoe krijgen we het resultaat van het CPR in de applicatie in een vorm die we kunnen gebruiken?

In de klassieke situatie hebben we een terminal, een computer en een applicatie:

```
VT100 <--> computer <--> applicatie
```

en als de applicatie een CPR aanvraag opstuurt, gaat dat via de computer naar de VT100, die antwoordt met de gevraagde informatie. Wat de applicatie dus moet doen is kijken welke tekens er van de terminal komen.

Met een terminal emulator is het wat ingewikkelder (zie bijvoorbeeld <http://www.linusakesson.net/programming/tty/index.php> voor een gedetailleerde beschrijving) maar we verwachten hetzelfde gedrag, het is tenslotte een terminal emulator.

In Forth kunnen we tekens ontvangen van het toetsenbord met het woord `key`, en de ontvangen code zichtbaar maken met `emit`, bijvoorbeeld met:  
`: echo-key-decimal BEGIN key . AGAIN ;`  
Als je dit uitvoert en bijvoorbeeld een Esc intoetst, dan verschijnt er inderdaad een 27.

Als we nu eens dit doen?

```
: CPR-echo CPR echo-key-decimal ;  
dan krijgen we: cpr-echo 27 91 50 51 59 49 48 82  
dat is dus esc [ 2 3 ; 1 0 R
```

Het lijkt erop dat het gelukt is om een CPR aanvraag op te sturen en het resultaat met de cursor positie te ontvangen.

Om hiervan een Cursor-positie woord van te maken (gforth):

```
: fout s" Onjuist Cursor Position Report data ontvangen" exception throw ;  
: cursorxy ( -- kolom-u rij-u )  
  27 emit ." [6n" \ vraag cursorpositie op  
  key 27 <> IF fout THEN key '[' <> IF fout THEN  
  0 BEGIN key dup ';' <> WHILE '0 - swap 10 * + REPEAT drop 1 - ( rij-u )  
  0 BEGIN key dup 'R <> WHILE '0 - swap 10 * + REPEAT drop 1 -  
  ( rij-u      kolom-u ) swap  
;
```

## Ook iets te melden?

Stuur uw ideeën, tips, programma's of projecten naar de redactie, zodat anderen daar kennis van kunnen nemen.

Bijdragen liefst per E-mail, Uiterlijk 1 week voor de bijeenkomst, naar [f.l.van.der.markt@kader.hcc.nl](mailto:f.l.van.der.markt@kader.hcc.nl)

Website van HCC!Forth:

<https://forth.hcc.nl>

