

De alternatieve rekenmachine

versie 3.0.2

Introductie

Deze rekenmachine is de meest recente versie van een rekenprogramma dat ik vele jaren geleden heb gemaakt als oefening voor de programmering van een (eenvoudige) parser in C. Voor de ‘kenners’, dat was nog in de tijd van MS-DOS, eind jaren tachtig. Een command-line tool die je rechtstreeks achter de DOS-prompt invoerde en die dan gelijk antwoord gaf:

```
C:\>reken 2+3
```

```
5
```

```
C:\>
```

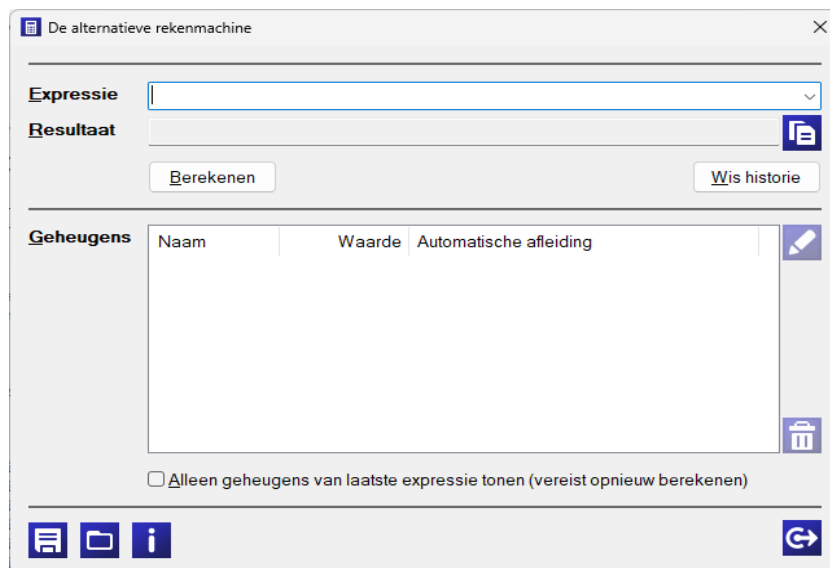
Later een versie gemaakt met een echt uiterlijk (‘interface’ in Computernederlands) met invoervak en resultaatvak.

Daarna opgevolgd door een eerste Windows-versie. En nog later zag de tweede Windows-versie het licht: uitgebreid met geheugens en later ook ‘formules’ (daarover meer, verder op in deze handleiding).

Intussen zijn we aanbeland bij versie 3. Met een nieuwe layout en een nieuwe binnenkant: de oude eigen parser is vervangen door de muParser. Dat is een open source-functiebibliotheek met een zeer uitgebreide en snelle parser.

Algemeen

Zoals de naam al aangeeft, is deze rekenmachine net iets anders dan de meeste rekenmachines:



Bij Expressie voer je de opdracht in die je uitgerekend wilt hebben. Na een druk op Enter, of een klik op Berekenen wordt de berekening uitgevoerd en het antwoord getoond bij Resultaat.

Bij het invoeren van de expressie mogen spaties gebruikt worden voor de leesbaarheid:

<i>expressie</i>	<i>resultaat</i>
2+3	5
2 + 3	5

Spaties ín een getal zijn (natuurlijk) niet toegestaan:

<i>expressie</i>	<i>resultaat</i>
------------------	------------------

23 + 11
2 3 + 11

34
foutmelding

Als decimaalteken zijn zowel de punt als de komma toegestaan:

<i>expressie</i>	<i>resultaat</i>
2 * 2,5	5
2 * 2.5	5

Invoer met een scheidingsteken voor de duizendtallen wordt niet ondersteund.



Met deze knop kopieer je het resultaat van een berekening naar het klembord.

Als je de Shift-toets ingedrukt houdt als je op deze knop klikt, wordt ook de ingevoerde expressie op het klembord gekopieerd:

Normaal:

<i>expressie</i>	<i>resultaat</i>	<i>naar klembord</i>
23 + 11	34	„34“

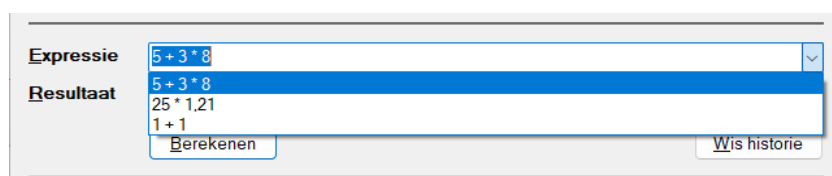
Met Shift-toets:

<i>expressie</i>	<i>resultaat</i>	<i>naar klembord</i>
23 + 11	34	„23 + 11 = 34“

De ingevoerde expressie blijft na de berekening zichtbaar, zodat je ook na een berekening nog kunt zien wat je hebt gedaan en een nieuwe berekening kunt uitvoeren op basis van de vorige.

Daarnaast wordt een lijst bijgehouden van de uitgevoerde berekeningen: de historie.

Je kunt door deze lijst met berekeningen bladeren met de verticale pijltjestoetsen (als Expressie de invoerfocus heeft), of de lijst als geheel tonen met toets F₄ of door te klikken op het pijltje aan de rechterkant van het expressieveld. Zo kunnen eerdere berekeningen makkelijk herhaald worden, al dan niet na aanpassing.



Met de knop Wis historie maak je de historie-lijst leeg.

Hoe komen we van die onvoldoendes af...

Een vreemde titel voor een hoofdstuk. Oudgedienden, zoals ikzelf, zijn grootgebracht met het ezelsbruggetje ‘Meneer van Dalen wacht op antwoord’. Beide zinnen zijn bedoeld om te onthouden wat de volgorde voor het uitvoeren van berekeningen is. Meneer van Dalen was al niet helemaal correct en is intussen achterhaald. Deze rekenmachine volgt de internationale standaardgroepen:

Hoe	haakjes
komen we	kwadrateren (machtsverheffen) en worteltrekken
van die	vermenigvuldigen en delen
onvoldoendes af	optellen en aftrekken

<i>expressie</i>	<i>resultaat</i>
2 + 3 * 4	14

$$(2 + 3) * 4$$

20

Binnen elke groep zijn de berekeningen gelijkwaardig en worden van links naar rechts uitgevoerd.

Geheugens (variabelen)

Een kenmerkend verschil tussen de alternatieve rekenmachine en de meeste andere rekenmachines is het aantal geheugens: deze rekenmachine beschikt over 1000 geheugens! Elk geheugen kan met een eigen naam, door jou gekozen, gebruikt worden. Bijvoorbeeld:

<i>expressie</i>	<i>resultaat</i>
a = 10	10
b = 5	5
a * b	50

De naam van een geheugen/variabele begint altijd met een letter of '_' en kan daarna uit letters, cijfers en het teken '_' bestaan. Let op: de rekenmachine maakt wel onderscheid tussen hoofdletters en kleine letters!

<i>expressie</i>	<i>resultaat</i>
lengte = 10	10
waarde1 = 5	5
Waarde1 = 15	15
btw_laag = 9	9

Geheugens kunnen natuurlijk ook gebruikt worden in berekeningen/expressies:

<i>expressie</i>	<i>resultaat</i>
lengte = 10	10
breedte = 5	5
oppervlak = lengte * breedte	50

Alle geheugens die in gebruik zijn, worden met hun waarde getoond in de lijst Geheugens:

Geheugens	Naam	Waarde	Automatische afleiding
	a	10	
	b	5	
	breedte	5	
	btw_laag	9	
	lengte	10	
	oppervlak	50	
	waarde1	5	

De geheugens staan in deze lijst altijd in alfabetische volgorde, waarbij hoofdletters voorgaan op kleine letters:

Speicher	Name	Wert	Automatische Ableitung
	A	25	
	B	11	
	a	10	
	b	5	

Zodra in deze lijst een regel geselecteerd is, worden de knoppen rechts actief:



Bewerken: hiermee wordt de naam van het geheugen naar het expressie-vak gekopieerd. Voor een eenvoudige invoer wordt '=' al achter de naam van het geheugen gezet om eenvoudig een nieuwe waarde te kunnen invoeren.

Als bij een geheugen een afleiding actief is, wordt deze als geheel naar het expressie-vak gekopieerd om deze te kunnen bewerken.

Deze knop is alleen actief als exact één regel geselecteerd is.



Verwijderen: met deze knop wordt het geselecteerde geheugen uit de lijst verwijderd.

Deze knop is ook actief als meerdere regels geselecteerd zijn, om op een eenvoudige wijze ook meerdere geheugens te verwijderen.

☐ Alleen geheugens van laatste expressie tonen (vereist opnieuw berekenen)

Als er veel geheugens in gebruik zijn, kan de lijst lang en onoverzichtelijk worden.

Met het inschakelen van deze optie, worden alleen nog maar de geheugens getoond, die in de laatst uitgevoerde berekening zijn gebruikt.

Na het in- of uitschakelen van deze optie, moet wel opnieuw een berekening worden uitgevoerd om het resultaat zichtbaar te maken.

Automatisch opnieuw berekenen

Om het uitvoeren van herhaalde berekeningen, waarin steeds maar een deel van de berekening verandert, te vereenvoudigen, kent deze rekenmachine 'automatische afleidingen': expressies die de waarde van een geheugen berekenen en steeds - automatisch - worden uitgevoerd zodra er een nieuwe expressie is uitgevoerd.

De oplettende lezer heeft in de lijst met geheugens al een derde kolom gezien: Automatische afleiding. Dat is de plaats waar deze 'formules' getoond worden.

Deze formules zijn gelijk aan een gewone expressie waarmee een waarde aan een geheugen wordt toegekend, maar dan met het teken '@' ervoor:

<i>expressie</i>	<i>resultaat</i>
@oppervlak = lengte * breedte	50

Na deze expressie toont de geheugenlijst:

Geheugens			
	Naam	Waarde	Automatische afleiding
	a	10	
	b	5	
	breedte	5	
	btw_laag	9	
	lengte	10	
	oppervlak	50	oppervlak = lengte * breedte
	waarde1	5	

Als nu een nieuwe waarde wordt ingevoerd voor lengte of breedte, worde de waarde van oppervlak automatisch herberekend:

<i>expressie</i>	<i>resultaat</i>
breedte = 25	25

	lengte	10	
	oppervlak	250	oppervlak = lengte * breedte
	waarde1	5	

Dit is dus een beetje vergelijkbaar met cellen in een spreadsheet. Maar niet met celnummers maar met logische namen.

Let op: alle afgeleide geheugens worden na iedere berekening opnieuw, éénmalig, berekend in alfabetische volgorde. Dit kan er soms voor zorgen dat afleidingen met verouderde waarden worden uitgevoerd. Door een extra 'lege' expressie (bijvoorbeeld alleen het getal 0) te laten uitvoeren, worden de afleidingen herhaald zonder bij-effecten en zullen de resultaten wel correct zijn.

Rekenkundige operatoren

Naast de hierboven genoemde 'standaard' bewerkingen, kent deze rekenmachine ook 'logische operatoren'. Die worden zelden in rekenmachines gebruikt, maar vaak in programmeertalen. Verderop in deze handleiding zullen we zien dat je deze extra mogelijkheden binnen deze rekenmachine goed kunt inzetten. In afnemende prioriteit:

<i>operator</i>	<i>prioriteit</i>	<i>betekenis</i>
$\wedge$	8	machtsverheffen, x tot de macht y
$*$	7	vermenigvuldigen
$/$	7	delen
$+$	6	optellen
$-$	6	aftrekken
$<=$	5	kleiner dan, of gelijk aan
$>=$	5	groter dan, of gelijk aan
$!=$	5	niet gelijk aan
$==$	5	gelijk aan
$>$	5	groter dan
$<$	5	kleiner dan
$\&$	4	bitgewijze 'en'
$ $	3	bitgewijze 'of'
$\&\&$	2	logische 'en'
$ $	1	logische 'of'
$=$	0	toewijzing, gelijkstelling (*)

(*) dit is een speciale vorm van operator, omdat deze geen berekening uitvoert, maar een waarde toekent aan een geheugen/variabele.

Operatoren met gelijke prioriteit worden altijd van links naar rechts uitgevoerd.

Om de oplettende lezer van dienst te zijn: in de lijst is worteltrekken niet aanwezig. Dat komt omdat dit wordt berekend met de operator $\wedge$ voor machtsverheffen. Daarvoor is even een stukje wiskunde-uitleg nodig:

de y^e -graad wortel uit x is gelijk aan x tot de macht $1 / y$.

<i>expressie</i>	<i>resultaat</i>
$4 \wedge 2$	16
$16 \wedge (1/2)$	4
$16 \wedge 0,5$	4

De 2^e-graadswortel is de 'normale' wortel, of vierkantswortel. Dit principe werkt uiteraard ook voor andere waarden:

<i>expressie</i>	<i>resultaat</i>
$3 \wedge 3$	27
$27 \wedge (1/3)$	3
$27 \wedge 0,33$	2,96722201251651

Bijzondere operator

Alle bovenstaande operators zijn zogenoemde ‘binaire operators’. Binair, omdat ze steeds betrekking hebben op twee elementen: links en rechts van het symbool.

Deze rekenmachine kent ook een ‘ternary operator’: een operator die betrekking heeft op drie elementen:

<i>operator</i>	<i>betekenis</i>
? :	als-dan-anders

Ook hier is weer een vergelijking te maken met een spreadsheet. Daar is dit meestal de functie ‘als’:

spreadsheet	als(conditie; waarde 1; waarde 2)
deze rekenmachine	conditie ? waarde 1 : waarde 2

De betekenis daarvan is:

als de conditie waar is, heeft de expressie de ‘waarde 1’, anders de ‘waarde 2’.

‘Waarde 1’ en ‘waarde 2’ kunnen zelf uiteraard weer uitgebreide expressies zijn.

In een praktijkvoorbeeld: de verkoopprijs is de inkoopprijs plus 20%. Behalve als de inkoopprijs hoger is dan 200, dan bedraagt de marge 15%. Dat ziet er dan zo uit:

verkoopprijs = inkoopprijs > 200 ? inkoopprijs * 1,15 : inkoopprijs * 1,20

Door deze als-dan-operator te combineren met automatische afleidingen, kunnen complexe berekeningen eenvoudig op vele situaties worden toegepast.

Ingebouwde functies

Deze rekenmachine beschikt over de meeste gangbare (en enkele minder gangbare) functies die je in een rekenmachine mag verwachten.

<i>functie</i>	<i>aantal argumenten</i>	<i>betekenis</i>
sin	1	sinus
cos	1	cosinus
tan	1	tangens
asin	1	arcsinus
acos	1	arccosinus
atan	1	arctangens
sinh	1	hyperbolische sinus
cosh	1	hyperbolische cosinus
tanh	1	hyperbolische tangens
asinh	1	hyperbolische arcsinus
acosh	1	hyperbolische arccosinus
atanh	1	hyperbolische arctangens
log2	1	logaritme met grondtal 2
log10	1	logaritme met grondtal 10
log	1	logaritme met grondtal e (2.71828...) (natuurlijke logaritme)
ln	1	logaritme met grondtal e (2.71828...) (natuurlijke logaritme)
exp	1	e tot de macht x
sqrt	1	vierkantswortel
sign	1	teken: -1 als x < 0; 0 als x = 0; 1 als x > 0

rint	1	afronden naar dichtsbijzijnde gehele getal (wiskundig afronden)
abs	1	absolute waarde
min	variabel	kleinste waarde van alle argumenten
max	variabel	hoogste waarde van alle argumenten
sum	variabel	som van alle argumenten
avg	variabel	gemiddelde waarde van alle argumenten

Let op: voor alle goniometrische functies geldt dat ze rekenen in **radialen**, niet in graden!

Graden kunnen natuurlijk omgerekend worden naar radialen (rad):

$\text{rad} = \text{graden} * 2 * \pi / 360$

Ingebouwde constanten

De waarden voor pi (π) en e (getal van Euler, grondtal voor de natuurlijke logaritme) zijn ingebouwd in de rekenmachine en kunnen, net als geheugens, direct in berekeningen worden opgenomen:

$\text{omtrek} = \text{straal} * 2 * \pi$

De voorgedefinieerde waarden van deze constanten zijn:

π 3,141592653589793238462643

e 2,718281828459045235360287

Morgen gaan we verder...

...waar we gebleven waren.

Weer een cryptische titel voor een kort hoofdstuk. Waar het hier om gaat: bij het afsluiten van de rekenmachine worden

- de inhoud van de expressie,
 - de 30 meest recente expressies van de historie, en
 - alle geheugens met waarde en automatische afleiding
- in het Register van Windows opgeslagen.

En bij een volgend gebruik worden deze weer opgehaald en teruggezet.

Gevolg: de rekenmachine ziet er net zo uit als bij het vorige gebruik. Je kunt verdergaan waar je gebleven was.

Of bewaren voor later

Als je veel verschillende soorten berekeningen met geheugens maakt. Of meer expressies (berekeningen) wilt behouden dan het standaard maximum van 30, dan is er nog een extra mogelijkheid: het opslaan van alle elementen (expressie, historie en geheugens) in een bestand. En op een willekeurig moment weer teruglezen.

Daarmee is het ook mogelijk om meerdere sets aan expressies en geheugens te bewaren en later weer te gebruiken voor identieke situaties.



Opslaan. Hiermee kun je de hele inhoud van de rekenmachine (actuele expressie, historie en alle geheugens met automatische afleidingen) opslaan in een bestand.

Het standaard bestandsformaat is `cdf`, afgeleid van ‘calculator data file’. Dit is een eigen bestandsformaat waar je als gebruiker niets anders mee kunt doen dan weer inlezen in de rekenmachine (of weggooien, natuurlijk).

Alternatief kan de inhoud ook als `csv`-bestand worden opgeslagen. Dat resulteert in een tekstbestand waarin je zelf kunt gaan sleutelen. Maar wel op eigen risico.



Inlezen. Hiermee kun je een eerder opgeslagen bestand weer inlezen.

De expressie wordt vervangen door de expressie-inhoud uit het bestand. De historie en de geheugens worden aangevuld met de waarden uit het bestand.

Tot slot

Dan blijven er twee knoppen over, die nog niet besproken zijn:



Informatie. Hiermee wordt deze gebruikershandleiding getoond.



Uitgang. Hiermee verlaat je de rekenmachine. Dat kan natuurlijk ook met de `Esc`-toets of de standaard toetscombinatie `Alt+F4`.

Informatie

De alternatieve rekenmachine is geschreven door Cuno Wegman.

Dit programma mag je vrij kopiëren, verspreiden en gebruiken, zolang je geen wijzigingen in het programma zelf aanbrengt en er geen vergoeding voor vraagt.

Een terugmelding over het gebruik wordt op prijs gesteld.

De meest recente versie is altijd te vinden op de downloadpagina van mijn website, www.ceegee.nl. Bij vragen, wensen, opmerkingen, fouten, of andere meldingen: stuur een e-mail naar reken@ceegee.nl.

Dit programma is mede gebaseerd op [muParser](#).

Historie

2026-05-26 *versie 3.0.0 build 14*

Eerste variant van versie 3 die verspreid kan worden.

2026-06-18 *versie 3.0.1 build 16*

Keuze voor subset van geheugens toegevoegd.

2026-07-18 *versie 3.0.2 build 20*

- Rekenmachine meertalig gemaakt: Duits en Engels toegevoegd.
- Met ingedrukte Shift-toets worden invoer én resultaat op het klembord gezet.
- Knop toegevoegd voor raadplegen handleiding.