




## How it works?

1. **Enter** text and formulas into the **"Code"** box on the left.
2. Press **F5** or click  to **calculate**. The results will appear in the **"Output"** box on the right as a professionally formatted Html **report**.
3. Click  to **print** or  to **copy** the output.

You can also **export** it to **Html** , **PDF**  or **MS Word**  document.

## The language

Calcpad language includes the following elements (click an item to insert):

- Real numbers: digits **"0"** - **"9"** and decimal point **"."**;
- Complex numbers: **re ± imi** (e.g. **3 - 2i**);
- Variables:
  - Latin letters: **"a"** - **"z"**, **"A"** - **"Z"**;
  - Greek letters: **"α"** - **"ω"**, **"A"** - **"Ω"**;
  - digits: **"0"** - **"9"**;
  - comma: **" , "**;
  - prime symbols: **" ' "**, **" " "**, **" ' ' "**, **" ' ' ' "**;
  - special symbols: **" ∅ "**, **" ∅ "**, **" ° "**, **" 4 "**;
  - **"\_"** for subscript;

A variable name must start with a letter. Names are case sensitive.

- Operators:
  - "!"** - factorial;
  - "^"** - exponent;
  - " / "** - division;
  - " ÷ "** - force division bar;
  - " \"** - division;
  - "%"** - reminder;
  - "\*"** - multiplication;
  - " - "** - minus;
  - " + "** - plus;
  - " ≡ "** - equal to;
  - " ≠ "** - not equal to;
  - " < "** - less than;
  - " > "** - greater than;
  - " ≤ "** - less or equal;
  - " ≥ "** - greater or equal;
  - " = "** - assignment;

- Custom functions of type  $f(x; y; z; \dots)$ ;
- Built-in functions:
  - sign**( $x$ ) – sign of a number;
  - abs**( $x$ ) - absolute value/magnitude;
  - sin**( $x$ ) - sine;
  - cos**( $x$ ) - cosine;
  - tan**( $x$ ) - tangent;
  - csc**( $x$ ) - cosecant;
  - sec**( $x$ ) - secant;
  - cot**( $x$ ) - cotangent;
  - sinh**( $x$ ) - hyperbolic sine;
  - cosh**( $x$ ) - hyperbolic cosine;
  - tanh**( $x$ ) - hyperbolic tangent;
  - csch**( $x$ ) - hyperbolic cosecant;
  - sech**( $x$ ) - hyperbolic secant;
  - coth**( $x$ ) - hyperbolic cotangent;
  - asin**( $x$ ) - inverse sine;
  - acos**( $x$ ) - inverse cosine;
  - atan**( $x$ ) - inverse tangent;
  - atan2**( $x; y$ ) - the angle whose tangent is the quotient of  $y$  and  $x$ ;
  - acsc**( $x$ ) - inverse cosecant;
  - asec**( $x$ ) - inverse secant;
  - acot**( $x$ ) - inverse cotangent;
  - asinh**( $x$ ) inverse hyperbolic sine;
  - acosh**( $x$ ) - inverse hyperbolic cosine;
  - atanh**( $x$ ) - inverse hyperbolic tangent;
  - acsch**( $x$ ) - inverse hyperbolic cosecant;
  - asech**( $x$ ) - inverse hyperbolic secant;
  - acoth**( $x$ ) - inverse hyperbolic cotangent;
  - log**( $x$ ) - decimal logarithm;
  - ln**( $x$ ) - natural logarithm;
  - log\_2**( $x$ ) - binary logarithm;
  - exp**( $x$ ) - exponential function;
  - sqr**( $x$ ) or **sqrt**( $x$ ) - square root;
  - cbrt**( $x$ ) - cubic root;
  - root**( $x; n$ ) -  $n$ -th root;
  - round**( $x$ ) - round to the nearest integer;
  - floor**( $x$ ) - round to the lower integer;
  - ceiling**( $x$ ) - round to the greater integer;

- trunc**( $x$ ) - round to the nearest integer towards zero;
- re**( $x$ ) - the real part of a complex number;
- im**( $x$ ) - the imaginary part of a complex number;
- phase**( $x$ ) - the phase of a complex number;
- random**( $x$ ) - random number between 0 and  $x$ ;
- min**( $x; y; z...$ ) - minimum of multiple values;
- max**( $x; y; z...$ ) - maximum of multiple values;
- sum**( $x; y; z...$ ) - sum of multiple values =  $x + y + z...$ ;
- sumsq**( $x; y; z...$ ) - sum of squares =  $x^2 + y^2 + z^2...$ ;
- srss**( $x; y; z...$ ) - square root of sum of squares = **sqrt**( $x^2 + y^2 + z^2...$ );
- average**( $x; y; z...$ ) - average of multiple values =  $(x + y + z...)/n$ ;
- product**( $x; y; z...$ ) - product of multiple values =  $x \cdot y \cdot z...$ ;
- mean**( $x; y; z...$ ) - geometric mean = **n-th root**( $x \cdot y \cdot z...$ );
- if**(*cond; value-if-true; value-if-false*) - conditional evaluation;
- switch**(*cond1; value1; cond2; value2; ...; default*) - selective evaluation;
- take**( $n; a; b; c...$ ) - returns the n-th element from the list;
- line**( $x; a; b; c...$ ) - linear interpolation;
- spline**( $x; a; b; c...$ ) - Hermite spline interpolation.
- Comments: "**Title**" or '**text**' in double or single quotes, respectively. **HTML**, **CSS**, **JS** and **SVG** are allowed.
  - Graphing and plotting:
 

**\$Plot** {  $f(x)$  @  $x = a : b$  } - simple plot;

**\$Plot** {  $x(t) | y(t)$  @  $t = a : b$  } - parametric;

**\$Plot** {  $f_1(x) \& f_2(x) \& ...$  @  $x = a : b$  } - multiple;

**\$Plot** {  $x_1(t) | y_1(t) \& x_2(t) | y_2(t) \& ...$  @  $x = a : b$  } - multiple parametric;

**\$Map** {  $f(x; y)$  @  $x = a : b \& y = c : d$  } - 2D color map of a 3D surface;

*PlotHeight* - height of plot area in pixels;

*PlotWidth* - width of plot area in pixels;

*PlotStep* - grid size for map plotting.
  - Iterative and numerical methods:
 

**\$Root** {  $f(x) = const$  @  $x = a : b$  } - root finding for  $f(x) = const$ ;

**\$Root** {  $f(x)$  @  $x = a : b$  } - root finding for  $f(x) = 0$ ;

**\$Find** {  $f(x)$  @  $x = a : b$  } similar to above, but  $x$  is not required to be a precise solution;

**\$Sup** {  $f(x)$  @  $x = a : b$  } - local maximum of a function;

**\$Inf** {  $f(x)$  @  $x = a : b$  } - local minimum of a function;

**\$Area** {  $f(x)$  @  $x = a : b$  } - adaptive Gauss-Lobatto numerical integration;

**\$Integral** {  $f(x)$  @  $x = a : b$  } – Tanh-Sinh numerical integration;

**\$Slope** {  $f(x)$  @  $x = a$  } - numerical differentiation;

**\$Sum** {  $f(x)$  @  $k = a : b$  } - iterative sum;

$\$Product \{ f(k) @ k = a : b \}$  - iterative product;

$\$Repeat \{ f(k) @ k = a : b \}$  - general inline iterative procedure;

*Precision* - relative precision for numerical methods [ $10^{-2}$ ;  $10^{-16}$ ] (default is  $10^{-12}$ )

- Program flow control:

Simple:

```
#if condition
    your code goes here
#end if
```

Alternative:

```
#if condition
    your code goes here
#else
    some other code
#end if
```

Complete:

```
#if condition1
    your code goes here
#else if condition2
    your code goes here
#else
    some other code
#end if
```

You can add as many "**#else if**"s as needed, but only one "**#else**". You can omit any of them.

- Iteration blocks:

Simple:

```
#repeat number of repetitions
    your code goes here
#loop
```

With conditional break:

```
#repeat number of repetitions
    your code goes here
    #if condition
        #break
    #end if
    some more code
#loop
```

- Modules and macros/string variables:

Modules:

```
#include filename - include external file (module);
#local - start local section (not to be included);
#global - start global section (to be included);
```

Inline string variable:

```
#def variable_name$ = content
```

Multiline string variable:

```
#def variable_name$
```

```
content line 1
```

```
content line 2
```

```
...
```

```
#end def
```

Inline string macro:

```
#def macro_name$(param1$; param2$; ...) = content
```

Multiline string macro:

```
#def macro_name$(param1$; param2$; ...)
```

```
content line 1
```

```
content line 2
```

```
...
```

```
#end def
```

- Output control:

**#hide** - hide the report contents;

**#show** - always show the contents (default);

**#pre** - show the next contents only before calculations;

**#post** - show the next contents only after calculations;

**#val** - show only the final result, without the equation;

**#equ** - show complete equations and results (default);

**#noc** - show only equations without results (no calculations);

Each of the above commands is effective after the current line until the end of the report or another command that overwrites it.

- Units for trigonometric functions: **#deg** - degrees, **#rad** - radians, **#gra** – grades;

- Separator for target units: **|**;

- Return angles with units: *ReturnAngleUnits* = 1;

- Angle units (dimensionless): °, ', ", *deg*, *rad*, *grad*, *rev*;

- Metric units (SI and compatible):

Mass: *g*, *hg*, *kg*, *t*, *kt*, *Mt*, *Gt*, *dg*, *cg*, *mg*, *µg*, *Da*, *u*;

Length: *m*, *km*, *dm*, *cm*, *mm*, *µm*, *nm*, *pm*, *AU*, *ly*;

Time: *s*, *ms*, *µs*, *ns*, *ps*, *min*, *h*, *d*;

Frequency: *Hz*, *kHz*, *MHz*, *GHz*, *THz*, *mHz*, *µHz*, *nHz*, *pHz*, *rpm*;

Velocity: *kmh*;

Electric current: *A*, *kA*, *MA*, *GA*, *TA*, *mA*, *µA*, *nA*, *pA*;

Temperature: °C, Δ°C, *K*;

Amount of substance: *mol*;

Luminous intensity: *cd*;

Area: *a*, *daa*, *ha*;

Volume: *L, mL, cL, dL, hL*;

Force: *dyn N, daN, hN, kN, MN, GN, TN, gf, kgf, tf*;

Moment: *Nm, kNm*;

Pressure: *Pa, daPa, hPa, kPa, MPa, GPa, TPa, dPa, cPa, mPa,  $\mu$ Pa, nPa, pPa, bar, mbar,  $\mu$ bar, atm, at, Torr, mmHg*;

Energy work: *J, kJ, MJ, GJ, TJ, mJ,  $\mu$ J, nJ, pJ, Wh, kWh, MWh, GWh, TWh, cal, kcal, erg, eV, keV, MeV, GeV, TeV, PeV, EeV*;

Power: *W, kW, MW, GW, TW, mW,  $\mu$ W, nW, pW, hpM, ks, VA, kVA, MVA, GVA, TVA, mVA,  $\mu$ VA, nVA, pVA, VAR, kVAR, MVAR, GVAR, TVAR, mVAR,  $\mu$ VAR, nVAR, pVAR*;

Electric charge: *C, kC, MC, GC, TC, mC,  $\mu$ C, nC, pC, Ah, mAh*;

Potential: *V, kV, MV, GV, TV, mV,  $\mu$ V, nV, pV*;

Capacitance: *F, kF, MF, GF, TF, mF,  $\mu$ F, nF, pF*;

Resistance:  *$\Omega$ , k $\Omega$ , M $\Omega$ , G $\Omega$ , T $\Omega$ , m $\Omega$ ,  $\mu\Omega$ , n $\Omega$ , p $\Omega$* ;

Conductance: *S, kS, MS, GS, TS, mS,  $\mu$ S, nS, pS,  $\mathcal{U}$ , k $\mathcal{U}$ , M $\mathcal{U}$ , G $\mathcal{U}$ , T $\mathcal{U}$ , m $\mathcal{U}$ ,  $\mu\mathcal{U}$ , n $\mathcal{U}$ , p $\mathcal{U}$* ;

Magnetic flux: *Wb, kWb, MWb, GWb, TWb, mWb,  $\mu$ Wb, nWb, pWb*;

Magnetic flux density: *T, kT, MT, GT, TT, mT,  $\mu$ T, nT, pT*;

Inductance: *H, kH, MH, GH, TH, mH,  $\mu$ H, nH, pH*;

Luminous flux: *lm*;

Illuminance: *lx*;

Radioactivity: *Bq, kBq, MBq, GBq, TBq, mBq,  $\mu$ Bq, nBq, pBq, Ci, Rd*;

Absorbed dose: *Gy, kGy, MGy, GGy, TGy, mGy,  $\mu$ Gy, nGy, pGy*;

Equivalent dose: *Sv, kSv, MSv, GSv, TSv, mSv,  $\mu$ Sv, nSv, pSv*;

Catalytic activity: *kat*;

- Non-metric units (Imperial/US):
 

Mass: *gr, dr, oz, lb, kip, st, qr, cwt, cwt<sub>UK</sub>, cwt<sub>US</sub>, ton, ton<sub>UK</sub>, ton<sub>US</sub>, slug*;

Length: *th, in, ft, yd, ch, fur, mi, ftm, cable, nmi, li, rod, pole, perch, lea*;

Speed: *mph*;

Temperature:  *$^{\circ}$ F,  $\Delta^{\circ}$ F,  $^{\circ}$ R*;

Area: *rood, ac*;

Volume (fluid): *fl\_oz, gi, pt, qt, gal, bbl, (dry) bu, fl\_oz<sub>UK</sub>, gi<sub>UK</sub>, pt<sub>UK</sub>, qt<sub>UK</sub>, gal<sub>UK</sub>, bbl<sub>UK</sub>, (dry) bu<sub>UK</sub>, fl\_oz<sub>US</sub>, gi<sub>US</sub>, pt<sub>US</sub>, qt<sub>US</sub>, gal<sub>US</sub>, bbl<sub>US</sub>, (dry) bu<sub>US</sub>*;

Force: *ozf, lbf, kipf, tonf, pdl*;

Pressure: *osi, osf psi, psf, ksi, ksf, tsi, tsf, inHg*;

Energy/work: *BTU, therm, therm<sub>UK</sub>, therm<sub>US</sub>, quad*;

Power: *hp, hpE, hpS*.