The Babylonian method for finding square roots - also known as Herons method

Reference: Algorithmen - von Hammurapi bis Gödel, 4. Auflage, Springer Spektrum, Wiesbaden 2016, 3.2 1

1 The standard procedure

The algorithm of Heron, in general, only gives approximate solutions. It it artificially stopped when the precision reached in the process reaches a certain threshold level, given by the value of a variable called epsilon.

```
(%i1) epsilon : 0.000000001 $;

(%i2) Heron(a) :=
        block([x : a],
          while abs(a-x*x) > epsilon do   x : (x+a/x)/2,
          x) $ ;

(%i3) Heron(5);        /*  For highest precision integer and fractional arithmetic is applied */
(%o3)  4870847
       -------
       2178309

(%i4) Heron(5.0);      /*  The input 5.0 signals that floating point arithmetic is wanted */
(%o4) 2.236067977499978

(%i5) Heron(25);
(%o5)  343368382031095922873155864089 7
       ---------------------------------
       686736764054813148116827907533

(%i6) Heron(25.0);      /*  an effect of floating point arithmetic */
(%o6) 5.000000000053722
```

2 Some experiments with number formats and precision

```
(%i7) fpprec : 100;     /* set precision to 100 */
(fpprec) 100

(%i8) epsilon :  bfloat(1.0e-100) ;             /* set epsilon to high value */  ;
(epsilon)
      1.0000000000000000199918998026[45 Ziffern]55543467617676288613292 99b-100

(%i9) Heron(bfloat(25));
(%o9) 5.0b0
```

(%i10) Heron(bfloat(25.0));
(%o10) 5.0b0

(%i11) %^2;
(%o11) 2.5b1

(%i12) Heron(28); /* very long integers are not fully displayed */
(%o12)
420201030217856789823046486169[758 Ziffern]082255392129985727377479940097
/
794105304721133035682668109886[757 Ziffern]693488613714378385682910352384

3 Enforce complete display of long integers

(%i13) set_display(ascii) $ /* display integers in full length - mind the division sign / */

(%i14) Heron(28);
(%o14) 42020103021785678982304648616990686401445444313142507583121065358627224
4713078412848250863850549133174102648421437756026173559421738837980733505269 75
3070686919551345793881056865382858077880032154329522873683190524337068976375 38
2856099355460650625663333189869313676929592567685078938824056386796280323143 51
8559800324284529204282036465524207254599334945631862726573807593890100090739 22
0550198175013816653500600641513261371616548110533885083128474898296106075758 92
9275161990967230813897706044567054442690723919937238689747957313806806610925 37
6226847890142276423473200573720326200320200277599222467928488792048975320249 77
0351336767529427833466626044551839997217328090050024695066328379748549190926 48
3805490016877691293396674215295865806761540243477752807916751708384342907595 85
2101985276019460822553921299857273774799400 97/794105304721133035682668109886 33
9213919635360624973882891451864072756828040379729430525615118285150021406966 52
2204151357180172517294678933871690378039606958244720845220089509091675508923 42
0779768183143420299676009261056453117095609367636677161512150021439508792957 94
4047236269853523681814467259344069200950459578303650143054783824339404566330 09
8766180250427218316541530862514149676538498939967662814786836064323974608054 43
5836801522762732152088427664011067030691669031450120867372294970584538181332 78
1704102808341250496362495855263754155825099701365351301978165972894401649310 63
7201547930427942079659097472875349147695046813519105342222642284120358113632 74
0553352308080937573671224595813665292999040626690382948527312998174120938284 37
8022440551209471783986475184930205110361373766227169169348861371437838568291 03
52384

(%i15) Heron(bfloat(28.0));
(%o15) 5.291502622129181181003231507278520851420518366164900360736668918402137
6464605672555207857729490872 21b0

(%i16) %^2; /* square the last output */
(%o16) 2.8b1

4 Several output types

```
(%i17) Heron_2(a) :=          /*  output type:  [fraction,  floating point number, bigfloat]  */
         block([x : a],
           while abs(a-x*x) > epsilon do x : (x+a/x)/2,
           [x, float(x), bfloat(x)] ) $ ;

(%i18) Heron_2(5);
                  31683700840009422215077673848376823600642097148698060 7
(%o18) [-----------------------------------------------,
                  14169381771405651323470996587541191965770779495819986 7
2.23606797749979, 2.23606797749978969640917366873127623544061835961152572427 08\
9724541052092563780489941441440837878227 5b0]

Some checks
   % gives the last output
    ^2 operates on every element of the output list

(%i19) %^2;
(%o19) [10038568989192137668875423999282625670487962768318190151509939861346 56\
1888480697130403512194736890559408844 9/20077137978384275337750847998565251340 9\
7592553663638030301987972269312377696139426080702438947378111881768 9,
5.000000000000001, 5.0b0]

Some checks with the same results using map and lambda

(%i20) map(lambda([x], x^2), Heron_2(5));
(%o20) [10038568989192137668875423999282625670487962768318190151509939861346 56\
1888480697130403512194736890559408844 9/20077137978384275337750847998565251340 9\
7592553663638030301987972269312377696139426080702438947378111881768 9,
5.000000000000001, 5.0b0]
```