Models of Reciprocal Powering: Mathematics and Software

Ruben Preiss
ruben.preiss@udp.cl
Institute of Basic Sciences, Faculty of Engineering
Universidad Diego Portales.
Santiago, Chile

Gonzalo Riera
griera@mat.puc.cl
Faculty of Mathematics
Pontificia Universidad Católica de Chile
Santiago, Chile

Mauricio Herrera
maherrer2005@gmail.com
Institute of Basic Sciences, Faculty of Engineering
Universidad Diego Portales
Santiago, Chile

Carlos Ruz
carlosruzl@gmail.com
Institute of Basic Sciences, Faculty of Engineering
Universidad Diego Portales
Santiago, Chile

Abstract: Reciprocal Powering is a product of interactive methodologies when implementing mathematics and software, for instance, Maple, The Geometer's Sketchpad or calculators as FX-2.0 or Classpad300, in mathematical research as well as in mathematical education. This involves a special way of undertaking mathematics and education as a simultaneous process of knowledge production together with teaching-learning process. The idea is to encourage the obtaining of new results, to improve thinking approaches and to optimize software through the development of special mathematical results and programming activities during the process of teaching-learning based on situations and circumstances originated on one hand by mathematics and on the other hand by introduction of computational software that involves propitious and necessary ways of reciprocal powering among both areas through the interaction and intercommunication of necessities and its solutions. Those methodologies are strongly related with experimental activities involved in the way of thinking and making research in mathematics as well as in the process of teaching-learning in mathematical education in order to complement an era deeply influenced by emphasis in formal proof as the most important feature of mathematical thinking and that had produced a way of creating and teaching mathematics with rigorous stages on how to teach, discover and develop mathematics following a particular order: (1) Formalization of theories; (2) Presentation of formal deduction; (3) Explanations including informal deduction; (4) Analysis to communicate what was involved in the deduction; and (5) Visualization given through examples. However, implementation of software involves new mechanisms of mathematics-software-related-thinking and own ways for creation of knowledge, involving the need to introduce new methodologies of research as well as in the process of teaching-learning that includes the recreation of some old methodologies that has been left because of the impossibility of carrying them ahead with efficiency due to the absence of computers or calculators. In using calculators or software, we need to reverse the order of the stages if our goal is getting better results in mathematics as well as in mathematical education by using technology. Following a suggestion done for geometry thinking by the Dutch mathematical

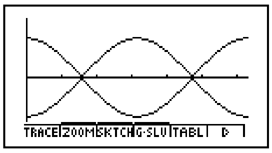
educators Pierre van Hiele and Dina van Hiele-Geldof and expanding it to other branches of mathematics the stages should be: (1) Visualization through particular cases and specific situations; (2) Analysis to obtain creative ideas that may lead to a generalization; (3) Informal deduction trying to get a generalized proof; (4) Obtainment of a first level formal deduction and (5) Review of the formal deduction to assure rigor. We propose a way of implementing technology with attainment of knowledge and development of mathematical thinking as a way to power mathematics and technology reciprocally. In our exposition we will give examples based on graphic, symbolic and programming situations that had led to reciprocal powering of mathematics and software, in particular with calculators.

1. Introduction

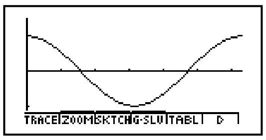
Our goal is to find a model for implementing mathematics with technology introducing a different teaching-learning process based on circumstances originated on one hand by *mathematics* and on the other hand by *computational software* (calculators) that should involve a reciprocal powering among both areas through interaction and intercommunication of necessities and its solutions. We call it *reciprocal powering. Reciprocal powering* is the product of necessities raised when implementing interactive methodologies and can be used as a special way of undertaking mathematics teaching as well as in production of knowledge in mathematics.

1. Powering mathematical concepts in Trigonometry through visualization with calculators.

Let's consider the very known formula $\sin^2 x + \cos^2 x = 1$ from where it is usually deduced $\cos x = \pm\sqrt{1-\sin^2 x}$. How can be verified this identity through a graphic way using a calculator? At fist glance this expression looks to correspond to two relations $\cos x = \sqrt{1-\sin^2 x}$ and $\cos x = -\sqrt{1-\sin^2 x}$. But when we enter both formulae to obtain a graphic in the interval $0 \le x \le 2\pi$ we obtain:



which is clearly different from cosine function graph in the same interval:



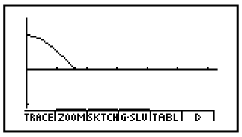
Therefore, according to the graph, the identity $\cos x = \pm\sqrt{1-\sin^2 x}$ is not valid. So, where is the mistake? It is known that cosine is positive in the first and in the fourth quadrant and negative in the second and third quadrant. That implies that the formula that relates cosine with sine is a *piecewise formula.* Therefore the way of writing the identity correctly should be:

$$\cos x = \begin{cases} \sqrt{1-\sin^2 x}\ , \text{ if } & 0 \le x \le \frac{\pi}{2} \\ -\sqrt{1-\sin^2 x} \text{ if } & \frac{\pi}{2} < x \le \frac{3\pi}{2} \\ \sqrt{1-\sin^2 x} \text{ if } & \frac{3\pi}{2} < x \le 2\pi \end{cases}$$

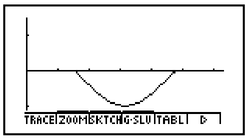
And if a calculator does not have piecewise functions commands (as Casio FX-2.0 Plus) then we can try to express it by including the graph of three relations:

$$Y1 = \sqrt{1-\sin^2 x}\ ,\ [0\ ,\ \pi/2\ ]$$
$$Y2 = -\sqrt{1-\sin^2 x}\ ,\ [\pi/2,\ 3\pi/2\ ]$$
$$Y3 = \sqrt{1-\sin^2 x}\ \ ,\ [3\pi/2\ ,\ 2\pi]$$

The graph for the first function will be:

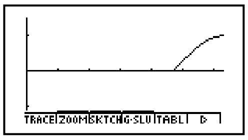


The graph of the second one will be:

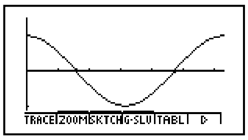


And the graph for the third function will be

:



By graphing altogether we will obtain exactly the very same graph of cosine:



It is important to state that all the classical formulae of trigonometry that involves square roots should be rewritten correctly. For instance, the following identity, in a first cycle:

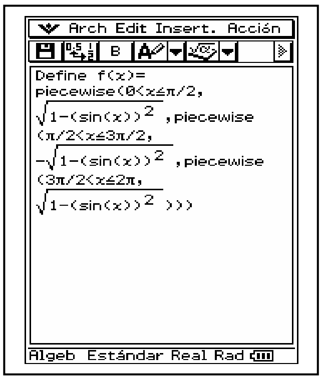
$$\sin\frac{x}{2} = \pm\sqrt{\frac{1-\cos x}{2}}$$

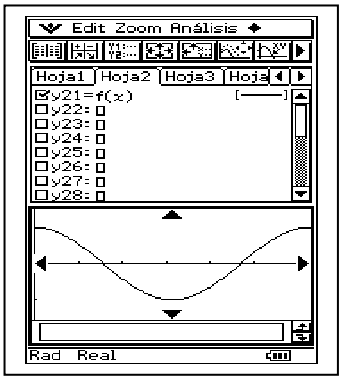
should be rewritten as

$$\sin\frac{x}{2} = \begin{cases} \sqrt{\frac{1-\cos x}{2}} & \text{si}\ \ x \in [\ 0\ ,\ 2\pi\ ] \\ -\sqrt{\frac{1-\cos x}{2}} & \text{si}\ \ x \in [\ 2\pi\ ,\ 4\ \pi\ ] \end{cases}$$

and note that we need an interval as [0 , 4 π] in order to obtain a complete first cycle, because: $0 \le \frac{x}{2} \le 2\pi \Rightarrow 0 \le x \le 4\pi$ and $\sin\frac{x}{2} \ge 0$ for $0 \le \frac{x}{2} \le \pi$. Therefore $\sin\frac{x}{2} \ge 0$ for $0 \le x \le 2\pi$ and $\sin\frac{x}{2} \le 0$ for $\pi \le \frac{x}{2} \le 2\pi$. Therefore $\sin\frac{x}{2} \le 0$ for $2\pi \le x \le 4\pi$.

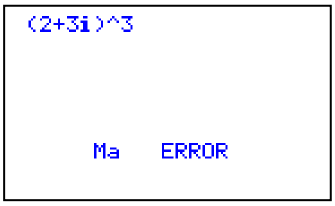
If we could work with a more complete calculator as Classpad 300, then we can use a description for a *function in pieces* by entering the command *piecewise*. Then the syntax for the identity between sine and cosine should be (it looks as a logical challenge):

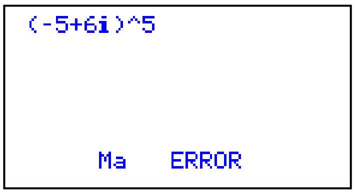


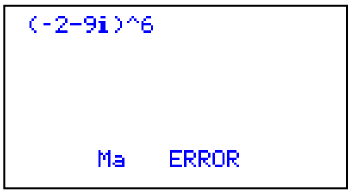


2. Powering calculator's software performance by creating and optimizing programming activities.

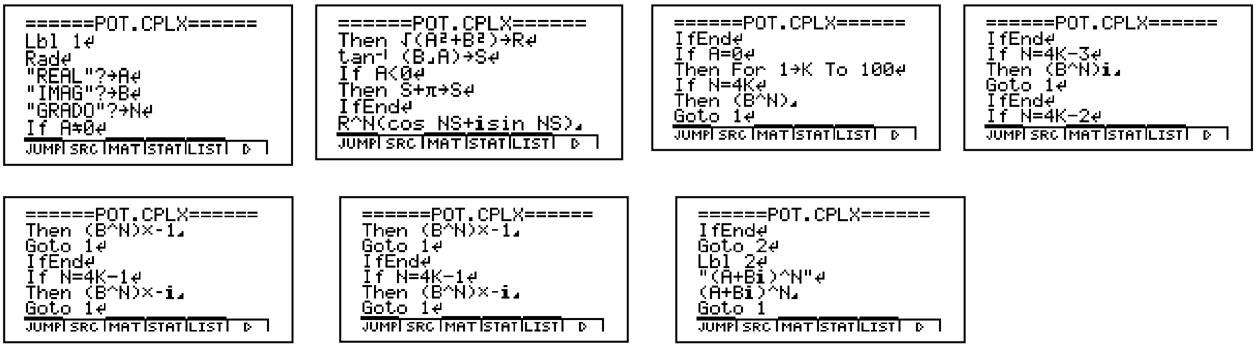
Programming is the necessary skill to be developed in this century. His presence in calculators has influenced strongly the educational process. On one side it has facilitated formalization in development of new ways of representation knowledge and on the other side it has raised the necessity of generating and developing new intellectual skills. The necessary stages for process of internalizing programming skills are the following: (1) Handle of symbolical logic language of calculators software; (2) Deep knowledge of mathematical relational concepts involved in programming; (3) Handle of a meta-language correlating mathematics and software; (4) Particular and detailed (numerical) checking previous of a process generalization; (5) Interpretation of model to be programmed and his interrelation with logical forms of software and optimization of their process. Our first need of introducing programming with calculators happened in 1997 when we tried to obtain with a Casio CFX-9850 G Plus calculator the power of a complex number. We obtained as an answer a "mathematical error":







After several attempts we realized that calculator Casio CFX-9850 G Plus was not designed to obtain powers of complex numbers although it was able to obtain the basic four operations with complex numbers. We had two choices: (1) Accept the machine limitation and continue calculations by hand, or (2) Try a bigger and more interesting challenge: Help the machine to surpass his own limitation, powering it with his own capacity of programming performance. But powering the calculator implied challenges not for the machine but for the students as users. It was necessary to start the process based on the five stages explained before, in order to power the calculator and get the way to obtain by programming way, powers of complex numbers: a) Knowledge of De Moivre's theorem ; b) Creation of a special subroutine for the case of powering pure imaginary numbers; because it was mathematically incorrect to try to calculate an angle ϑ in $\arctan(b/a) = \vartheta$ when considering the case a = 0; c) Need of several logic tautologies to simplify and depurate the extreme length of the program that involved a delayed running program ; d) Correction of initial mistakes in the checking stage related to the need of creating a small program that would allow the machine “to know” which of the two solutions should be considered when we were searching ϑ in the equation $\arctan(b/a) = \vartheta$. The following program was obtained by undergraduate engineering students of a first semester algebra course and can be used for a and b rational numbers with a Casio CFX-9850-G Plus calculator as well as with a Casio Algebra FX-2.0:



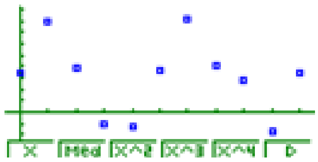
As an additional contribution, those students constructed a program that was able to find the *root of a complex number* as a sub product of the former program which can run with Casio CFX-9850 G Plus as well as with Casio Algebra FX-2.0 Plus.

3. Powering calculator’s software performance by developing mathematics and creating programming activities: Periodic regressions with a calculator.

During 1997 and 1998 we were working with an old version of Casio CFX-9850 G Plus calculators in order to implement the use of calculators in mathematical areas of engineering careers. And we realized that one of the frequent situations confronted by students who were just beginning their studies of calculus in a first year of college, before a course in statistics, is the common type of problems they used to study. Practically all of them departed from formulae which are not a product of a real or practical situation. Our idea was to depart of situations that are data of real life. With calculators this looked to be something possible, because with them we could decide to depart from values or data and then list them in order to obtain an automatic regression with the calculator. However we found out a problem with calculators in that moment. Although they were able to get linear, quadratic, cubic, fourth, exponential and logarithmic regressions, they were not able to get periodic regression, something so frequent and necessary in real life (for instance: pulse, inhalation and exhalation when we breathe, etc). Our goal was to answer the following questions: (1) How it is possible to find a method to get periodical regressions ? (2) How can we utilize the power of a calculator as Casio CFX-9850 G Plus or a Casio FX-2.0 Plus to obtain a program for a periodical regression, the corresponding graph and check that it fits sufficiently the given points? To get our goal we had to power some results based on Theory of Fourier Analysis of Time Series.

3.1. Least squares estimation for a simple wave

Suppose we have some numerical values with some periodicity. As an example we could consider the magnitude of a variable star at midnight represented in a graph of points.



We would like to obtain the best possible fit of these values with a simple wave of the form:

$$\mu + R\cos(\omega t + \phi) \qquad\qquad f(t) = \mu + A\cos\omega t + B\mathrm{sen}\omega t$$

Given the period $\omega = \frac{2\pi}{n}$ we can find μ, A, B such that T is as small as possible, where:

$T = \sum_{t=0}^{n}(x_t - \mu - A\cos\omega t - B\sin\omega t)^2$ and x_t is the numerical value at time t. Differentiating we obtain:

$$\frac{\partial T}{\partial \mu} = -2\sum(x_t - \mu - A\cos\omega t - B\sin\omega t) = 0\,;$$

$$\frac{\partial T}{\partial A} = -2\sum \cos\omega t(x_t - \mu - A\cos\omega t - B\sin\omega t) = 0\,;$$

$$\frac{\partial T}{\partial B} = -2\sum \sin\omega t(x_t - \mu - A\cos\omega t - B\sin\omega t) = 0$$

This is a linear system of three equations and three unknowns:

$$n\mu + A\sum\cos\omega t + B\sum\sin\omega t = \sum x_t$$

$$(\sum\cos\omega t)\mu + A\sum\cos^2\omega t + B\sum\cos\omega t\sin\omega t = \sum\cos\omega t x_t$$

$$(\sum\sin\omega t)\mu + A\sum\cos\omega t\sin\omega t + B\sum\cos^2\omega t = \sum\sin\omega t$$

Using the Euler relation $e^{i\omega} = \cos\omega + i\,\mathrm{sen}\,\omega$ and his inverse:

$\cos\omega = \frac{1}{2}\{e^{i\omega} + e^{-i\omega}\}$; $\mathrm{sen}\,\omega = \frac{1}{2i}\{e^{i\omega} - e^{-i\omega}\}$ it is possible to prove the following formulae:

$$\sum\cos\omega t = \cos\left(\frac{n-1}{2}\omega\right)\frac{\sin n\omega/2}{\sin\omega/2}\,; \qquad \sum\sin\omega t = \sin\left(\frac{n-1}{2}\omega\right)\frac{\sin n\omega/2}{\sin\omega/2}\,;$$

$$\sum\cos^2\omega t = \frac{n}{2}\left(1 + \frac{\sin n\omega}{n\sin\omega}\cos(n-1)\omega\right); \qquad \sum\sin\omega t\cos\omega t = \frac{n}{2}\frac{\sin n\omega}{n\sin\omega}\sin(n-1)\omega\,;$$

$$\sum\sin^2\omega t = \frac{n}{2}\left(1 - \frac{\sin n\omega}{n\sin\omega}\cos(n-1)\omega\right)$$

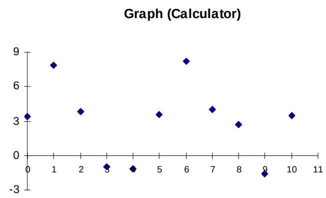
The former linear system can be solved and the wave can be found.

3.2 Approximation of periodic data with a Graphic Calculator.

We shall show how to find a wave that approximates the following set of data:

t	0	1	2	3	4	5	6	7	8	9	10
x_t	3,4	7,9	3,8	-1	-1,2	3,6	8,2	4	2,7	-1,6	3,5

If we plot the points, we obtain the following graph:



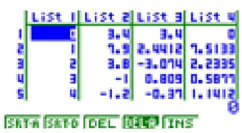
Since there are two peaks, the period is 5, so that $\omega = \frac{2\pi}{5}$, (n = 10). Here:

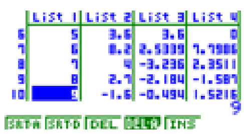
$$\sum \cos\omega t = \cos(\frac{18\pi}{10})\frac{\sin(10\frac{2\pi}{10})}{\sin(2\pi/5)} = 0; \qquad \sum \sin\omega t = 0;\ \sum \cos^2\omega t = 5;$$

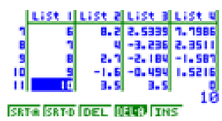
$$\sum \sin\omega t\cos\omega t = 0;\ \sum \sin^2\omega t = 5$$

The equations (*) are:

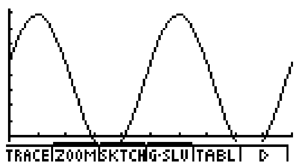
$10\mu = \sum x_t$; $\mu = \bar{x} = 3.33$; $5A = \sum_{t=0}^{10}(\cos(\frac{2\pi}{5}t))x_t$; $5A = 6,92 \Rightarrow A = 1,38$; $5B = \sum_{t=0}^{10}\sin(\frac{2\pi}{5}t)x_t$ $5B = 21,56 \Rightarrow B = 4,31$. Similar equations will appear if we have an integral number of times a period in the interval $\omega = \frac{2\pi}{m}$, $m = \frac{n}{k}$, $\sin n\omega/2 = \sin(k\pi) = 0$. Computing those values in the List Window List 1, List 2, List 3, List 4 of the calculator we have



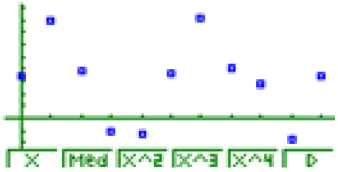




We find then the equation $f(t) = 3,33 + 1,38\cos\frac{2\pi}{5}t + 4,31\text{sen}\frac{2\pi}{5}t$ whose graph is:



compared with:



The former process can be generalized through a programming processing over lists in the calculator in such way that the calculator produces automatically the desired periodic regression. We show this program next:

```
======REG TRIG======
"NO.DS.="?→N↵
"PEAKS"?→K↵
N/K→M↵
2π/M→W↵
"W"↵
W◢
JUMP SRC MAT STAT LIST ▷
```

```
======REG TRIG======
W/2→H↵
cos (((N-1)/2)W)(sin
(NH))/sin H→P↵
"Σ(cos WT)"↵
P◢
sin (((N-1)/2)W)(sin
JUMP SRC MAT STAT LIST ▷
```

```
======REG TRIG======
(NH))/sin H→Q↵
"Σ(sin WT)"↵
Q◢
(N/2)(1+((sin NW)/(Ns
in W))cos ((N-1)W)→R↵
"Σ((cos WT)²)"↵
JUMP SRC MAT STAT LIST ▷
```

```
======REG TRIG======
R◢
(N/2)((sin NW)/(Nsin
W))sin ((N-1)W)→S↵
"Σ(sin WTcos WT)"↵
S◢
(N/2)(1-((sin NW)/(Ns
JUMP SRC MAT STAT LIST ▷
```

```
======REG TRIG======
in W))cos ((N-1)W)→T↵
"Σ((sin WT)²)"↵
T◢
Sum List 2→Z↵
(cos (W×List 1))×List
 2→List 3↵
JUMP SRC MAT STAT LIST ▷
```

```
======REG TRIG======
(sin (W×List 1))×List
 2→List 4↵
Sum List 3→G↵
Sum List 4→F↵
[[N,P,Q,Z][P,R,S,G][Q
,S,R,F]]◢
JUMP SRC MAT STAT LIST ▷
```

```
======REG TRIG======
"DET.PRIN"↵
Det [[N,P,Q][P,R,S][Q
,S,R]]→E◢
"U"↵
Det [[Z,P,Q][G,R,S][F
,S,R]]/E→U◢
JUMP SRC MAT STAT LIST ▷
```

```
======REG TRIG======
"A"↵
Det [[N,Z,Q][P,G,S][Q
,F,R]]/E→A◢
"B"↵
Det [[N,P,Z][P,R,G][Q
,S,F]]/E→B◢
JUMP SRC MAT STAT LIST ▷
```

```
======REG TRIG======
ClrGraph↵
ViewWindow (Min(List
1),Max(List 1),1,Min(
List 2),Max(List 2),1
↵
Y=Type↵
JUMP SRC MAT STAT LIST ▷
```

```
======REG TRIG======
↵
Y=Type↵
"U+Acos (WX)+Bsin (WX
)"→Y1↵
G SelOn 1↵
DrawGraph
JUMP SRC MAT STAT LIST ▷
```

4. An algorithm for calculation of π.

With aid of the limit $\lim_{\theta\to 0}\frac{\sin(\theta)}{\theta}=1$ it is possible to generate a basic procedure to determine the value of π. Together with this limit we will use also the following trigonometric relations:

$$\sin\left(\frac{x}{2}\right)=\sqrt{\frac{1-\cos(x)}{2}}\quad;\quad(0.1)\qquad\qquad \cos\left(\frac{x}{2}\right)=\sqrt{\frac{1+\cos(x)}{2}}\quad;\quad(0.2)$$

Taking $x=\frac{\pi}{3}$ we have $\cos\left(\frac{\pi}{3}\right)=\frac{1}{2}$ and introducing the notation:

$$x_i=\cos\left(\frac{\pi}{3\,2^i}\right)\;;\quad y_i=\sin\left(\frac{\pi}{3\,2^i}\right);\quad x_0=\frac{1}{2}. \tag{0.3}$$

We can construct the following recurrence formulae:

$$x_0=\frac{1}{2}\qquad y_{i+1}=\sqrt{\frac{1-x_i}{2}}\qquad x_{i+1}=\sqrt{\frac{1+x_i}{2}} \tag{0.4}$$

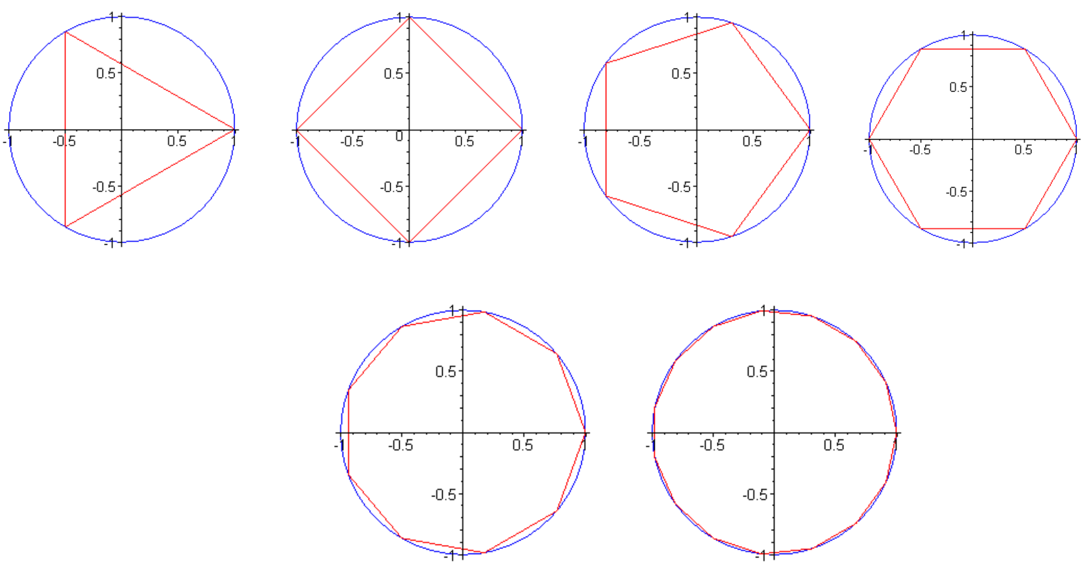
To calculate the value of π we can rewrite the limit $\lim_{x\to 0}\frac{\sin(x)}{x}=1$ in the following way:

$$\lim_{i \to \infty} \frac{\sin\left(\frac{\pi}{3\,2^i}\right)}{\frac{\pi}{3\,2^i}} = 1 . \qquad (0.5)$$

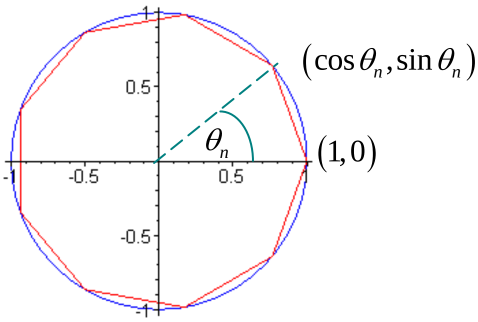
In this form the limit tells us that $3\,2^i \sin\left(\frac{\pi}{3\,2^i}\right)$ tends to π when i tends to ∞. According to the notation introduced in (1.3), the value of π can be calculated through the expression $3\,2^i y_i$, if we take a sufficient big number of i. In the following table we can find the values obtained after repeating the calculation ten times with the help of a simple program done in Maple. To avoid rounding errors we will take a sufficient big number of digits during the calculation.

Iteration	Value of π
1	3.0000000000000000000
2	3.1058285412302491482
3	3.1326286132812381966
4	3.1393502030468672043
5	3.1410319508905096368
6	3.1414524722854620456
7	3.1415576079118574430
8	3.1415838921483182961
9	3.1415904632280477600
10	3.1415921059992727998

This procedure has a simple geometric interpretation. It is based on the idea that the length (perimeter) of the polygon inscribed in a circle of radius one tend to the length of circle 2π when the number of sides of the polygon tends to infinity. The following figures done in Maple for $n = 3,4,5,6,10,15$ will show an idea of the process.



If we take the length of one of the inscribed sides of the polygon, as seen in the following picture we have:



$$d_n = \sqrt{(\cos\theta_n - 1)^2 + \sin^2\theta_n} = \sqrt{2(1-\cos\theta_n)} = 2\sin\frac{\theta_n}{2} \qquad (0.6)$$

For very big n we will have that n times the side d_n is approximately equal to the length of the circle $nd_n \approx 2\pi \Rightarrow n\cdot 2\sin\frac{\theta_n}{2} \approx 2\pi$. And therefore we have the relation:

$$n\cdot\sin\frac{\theta_n}{2} \approx \pi \qquad (0.7)$$

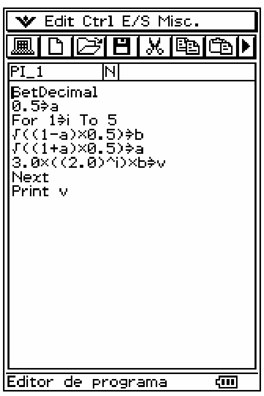
When we divide the circle in $3\cdot 2^i$ parts we have that $\theta_n = \frac{2\pi}{3\cdot 2^i}$. If we replace it in the relation (1.7), we will obtain:

$$3\cdot 2^i \cdot \sin\frac{2\pi}{2\cdot 3\cdot 2^i} \approx \pi \qquad (0.8)$$

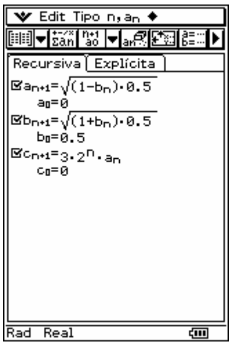
Simplifying we will have $3\cdot 2^i \cdot \sin\frac{\pi}{3\cdot 2^i} \approx \pi$, which is actually the relation we used to complete the former table.

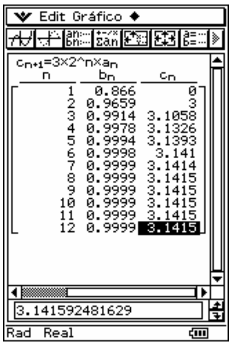
4.1 Implementation with Classpad300

With the help of the calculator we can implement this algorithm by constructing the following program:



However a more effective way can be obtained by using the following sequences obtaining the following results:





5. Powering calculator's software performance by using mathematics and creating programming activities:

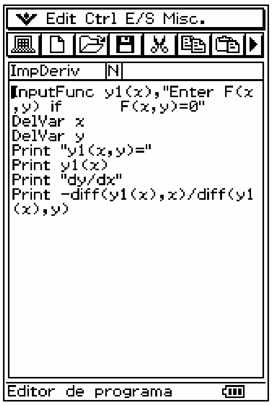
As is it known calculators do not usually work with implicit relations. It is important to note that ***diff*** gives usually derivatives that actually are partial derivatives each time when this command is used. Now let's suppose that we have an implicit expression. We can always write it as $F(x, y) = 0$.
Taking differential on both sides we will have: $dF = \frac{\partial F}{\partial x} dx + \frac{\partial F}{\partial y} dy = 0$ from where we can obtain:

$$\frac{dy}{dx} = -\frac{\frac{\partial F}{\partial x}}{\frac{\partial F}{\partial y}} \qquad (*)$$

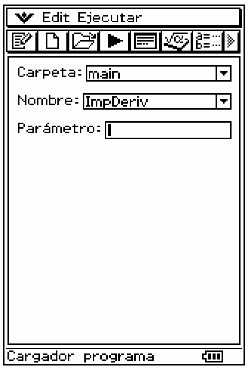
The formula (*) means that we can find derivatives of implicit expressions using the ***diff*** command, by obtaining partial derivatives and get the final result using (*).

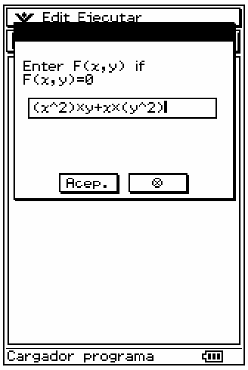
5.1 Implicit differentiation for calculators with symbolic programming.

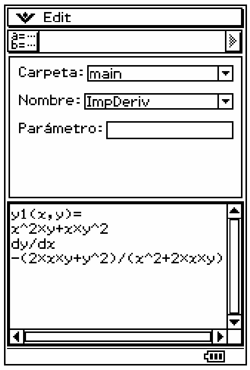
Using formula (*) we can construct a program for implicit differentiation for calculators as Classpad 300:



If we run the program for implicit differentiation for Classpad300 we will obtain for instance:

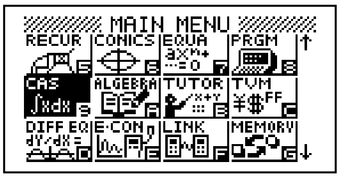


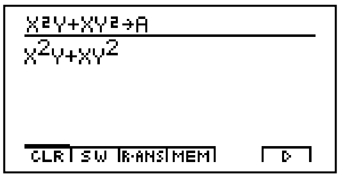


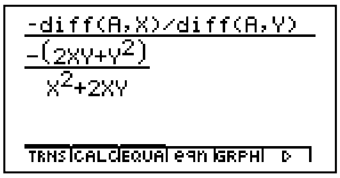


5.2 Implicit differentiation for calculators without symbolic programming.

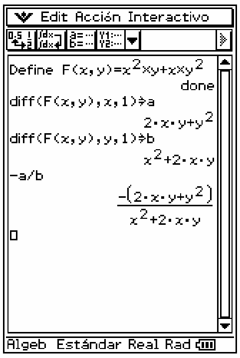
If we are working with calculators provided with a computer algebraic system but without symbolic programming as Casio Algebra FX-2.0 Plus, then we can enter the formula (*) each time we need it, as for instance:







or in the main menu of Classpad300:

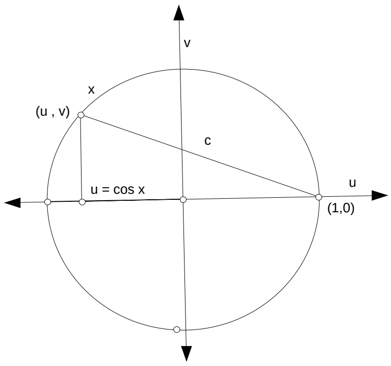


6. Powering a key feature of a calculator by developing mathematics and programming.

The following program for a Casio Algebra FX-2.0 as well as a Casio CFX-9850 GPlus optimizes the cosine feature. It is inspired in [1] and it can be used by entering degrees (program named as TCOSG type) as well as radians (program named as TCOSR type). The reason for having added number 16 to the name TCOSG16 will be understood after analyzing the program:

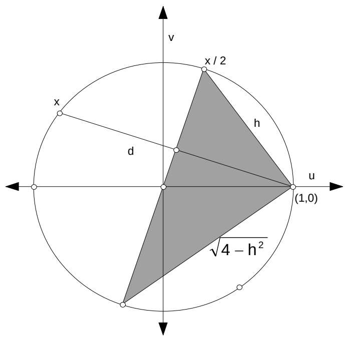
```
PROGRAM:   TCOSG16

Lbl 1 ↵
? → X ↵
Abs (X) → X ↵
X x π /180 → X ↵
XxX / 4294967296 → S ↵
For 1 → K To 16 ↵
S x (4 - S) → S ↵
Next ↵
1- S / 2
Goto 1
```

To understand how the program has been constructed we need two geometrics theorems. The first relates cosine x to the length of a chord associated with the arc x which can be seen in the following figure in which the following relations holds:



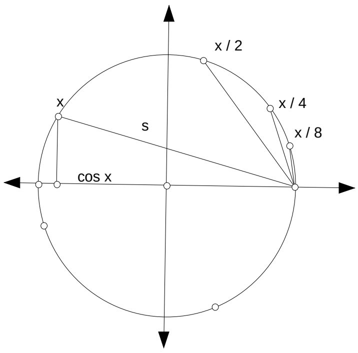
$$(u - 1)^2 + v^2 = c^2$$
$$u^2 + v^2 = 1$$
$$u = 1 - \frac{1}{2}c^2$$
$$\cos x = 1 - \frac{1}{2}c^2$$

The second theorem relates the length of a chord h to the length of a chord d of twice its arc which can be seen in the following figure and following relations holds:



$$A = \frac{1}{2} 2 \frac{d}{2}$$
$$A = \frac{1}{2} h \sqrt{4 - h^2}$$
$$d = h\sqrt{4 - h^2}$$
$$d^2 = h^2(4 - h^2)$$

With the following figure we can see how those theorems are applied to finding cosine x by the stated algorithm:

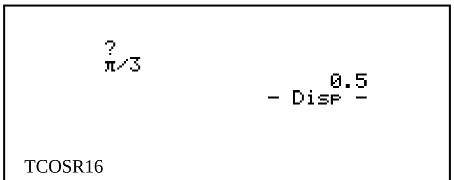


Notice in this last figure that each time we halve an arc, the chord length is closer to the arc length. In the algorithm the original arc x is halved 16 times; that is, divided by $2^{16} = 65536$. For that short an arc, clearly the chord is almost exactly equal in length and we take it to be equal. We square it (that is where the 4294967296 comes from the algorithm. $(h/65536)^2 = h^2/4294967296$) and use our second theorem sixteen times to double its length, noting that in the algorithm s represents both h^2 and d^2. When we are back to the chord length (squared) corresponding to the arc x, we use the first theorem to get cosine x. This time our s is c^2.

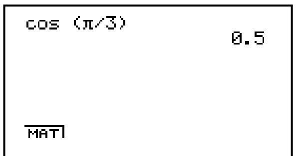
Before running the program, let's see a program adapted for radians:

```
PROGRAM:   TCOSR16

Lbl 1 ↵
? → X ↵
Abs (X) → X ↵
XxX  /  4294967296 → S ↵
For 1 → K To 16 ↵
S x (4 - S) → S ↵
Next ↵
1- S / 2
Goto 1
```

We will run now our program and compare it with the key cosine of the calculator for some values. For instance for $X = \dfrac{\pi}{3}$ we obtain:

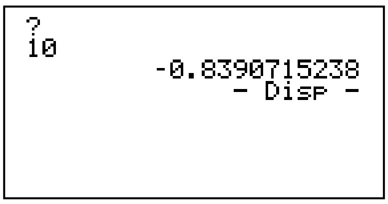


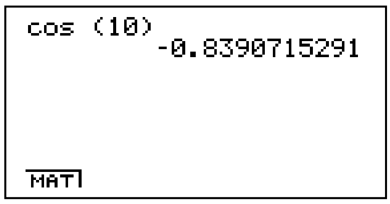
TCOSR16



KEY COS

But for a bigger value, results differ in the last two digits:



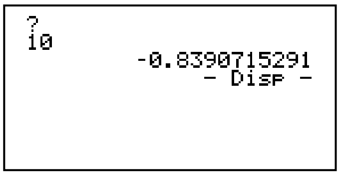


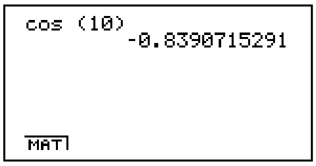
We construct then a similar program but with a bigger number of iterations:

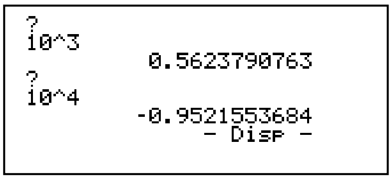
```
PROGRAM:  TCOSG50

Lbl 1 ↵
? → X ↵
Abs (X) → X ↵
X x π/180 → X ↵
XxX / (2^50)²  → S ↵
For 1 → K To 50 ↵
S x (4 - S) → S ↵
Next ↵
1- S / 2
Goto 1
```

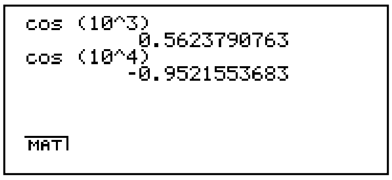
```
PROGRAM:  TCOSR50

Lbl 1 ↵
? → X ↵
Abs (X) → X ↵
XxX / (2^50)²  → S ↵
For 1 → K To 50 ↵
S x (4 - S) → S ↵
Next ↵
1- S / 2
Goto 1
```

We try again with values with for $X = 10^n$; with $n = 1,2,3........,10$:
Now results agree but running the program takes more time. And for bigger numbers last digit starts also to vary:





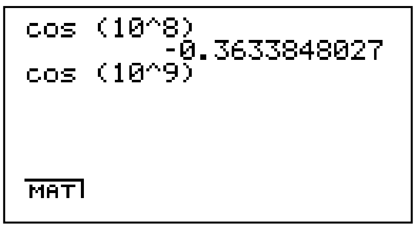




If we increase now to the maximum iteration accepted by the calculator (up to 166 iterations) the program in radians will be:

```
PROGRAM: TCOSR166

Lbl 1 ↵
? → X ↵
Abs (X) → X ↵
XxX / (2^166)²  → S ↵
For 1 → K To 166 ↵
S x (4 - S) → S ↵
Next ↵
1- S / 2
Goto 1
```

With this number of iterations we can get results with our program up to $\cos(10^{12})$ with results finally similar to Maple only up to the first decimal. But the key calculator collapse earlier, after $\cos(10^8)$!!



We can see the comparison of our program with Maple:

```
?
10^9
            0.8379514599
?
10^10
            0.8725454348
                 - Disp -
```

```
?
10^11
            0.3809482517
?
10^12
            0.7235914205
                 - Disp -
```

```
> evalf(cos(10^9));
                .8378871814
> evalf(cos(10^10));
                .8731196227
> evalf(cos(10^11));
                .3708477922
> evalf(cos(10^12));
                .7914463019
```

In this way we have powered our key calculator feature by relating theorems with programming activity.