

## Giới thiệu

- ✓ GCalc (General calculator) là chương trình có tác dụng như một máy tính cầm tay:
  - Gồm hơn 80 loại toán tử, tính được tất cả các biểu thức toán học thông thường, kể cả các biểu thức Logic. Cung cấp một số phép tính hữu dụng để tính ngày, giờ.
  - Giải các phương trình thông thường và tìm nghiệm gần đúng của phương trình một ẩn bất kỳ.
  - Vẽ đồ thị các hàm số sơ cấp.
  - Tính toán với ma trận
- ✓ Hãy thử bắt đầu chương trình bằng cách gõ một vài biểu thức vào ô Nhập một biểu thức. (mỗi lần nhập xong một biểu thức nhấn ENTER)

```
3 * sin( pi / 2 )
1:4 + 1:8
[ 4/2/1984 ] + 100
dow( [ 4/2/1984 ] )
canchi_y( 2002 )
bin( [100110] ) + 10
T -> ( T and F )
```

Mục đích của CT là giúp bạn tính toán tất cả những gì có thể một cách nhanh chóng nhất, dễ dàng nhất. Vì vậy, hãy bỏ một giờ tập sử dụng chương trình và bạn đã có trong tay một công cụ tính toán mạnh.

## Thông nhất quy ước

### **Biểu diễn một phân số:**

Toán tử **:** dùng để biểu diễn một phân số.

VD: muốn tính  $3/2 + 1/6 * 4/16$  viết là **3:2+1:6\*4:16**

### **Quy ước viết một Hàm :**

- Khi viết một hàm, phải có đóng mở ngoặc đơn bao quanh các tham số, và phải ngăn cách giữa các tham số bằng dấu phẩy.
- Tên hàm phải viết toàn bằng chữ thường. Viết chữ hoa chương trình sẽ không hiểu.

VD1: **ucln 30,5** là cách viết sai vì thiếu ngoặc bao quanh tham số.

hay **Ucln(30 5)** là cách viết sai vì thiếu dấu phẩy ngăn các tham số và có một chữ viết hoa trong tên hàm.

-> Cách viết đúng là **ucln(30,5)**

VD2: `bin[100110]` là cách viết sai vì thiếu dấu ngoặc đơn bao quanh tham số `[100110]`.

-> Cách viết đúng phải là `bin([100110])`

### **Quy ước viết một Hằng số:**

Hằng số phải được định nghĩa trong bảng quản lý hằng số (trừ các hằng đặc biệt `cdate`, `ctime`, `cdatetime`). Khi sử dụng hằng số, viết tên hằng phải chính xác cả về việc viết chữ hoa hay chữ thường.

Để bật bảng hằng số, nhấn nút có biểu tượng chữ C trên thanh công cụ.

### **Quy ước tính toán:**

1) Khi sử dụng một toán tử (hàm hoặc phép tính) mà đòi hỏi toán hạng nào đó phải là số nguyên, nếu bạn cố ý dùng toán hạng số thực vào đó thì chương trình sẽ tự động làm tròn toán hạng đó để tính toán đồng thời đưa ra lời cảnh báo ở cuối cửa sổ chính.

VD: phép chia lấy dư chỉ dùng với 2 toán hạng kiểu số nguyên. Nếu cố ý đánh biểu thức sau vào:

`7.8 mod 4.2`

thì chương trình sẽ tự làm tròn 7.8 thành 8; 4.2 thành 4 và trả lại kết quả của `8 mod 4` đồng thời gửi cảnh báo vào hộp danh sách cuối cửa sổ chính. Hãy thử mà xem.

2) Bạn có thể đổi kết quả trong hộp chứa kết quả sang các hệ cơ số khác bằng cách nhấn vào các nút chọn hình tròn ở dưới hộp chứa kết quả. Nhưng hãy dè chừng, nếu nội dung trong hộp kết quả không phải là số nguyên thì chương trình sẽ tự làm tròn kết quả trước khi đổi sang hệ cơ số mà bạn chọn

VD: hộp kết quả đang chứa số 2.6, nếu bạn nhấn vào nút tròn nhấn là `bin` (hoặc nhấn `Alt+2`), chương trình sẽ làm tròn 2.6 thành 3 rồi mới đổi sang hệ cơ số 2 và thu được kết quả là chuỗi kí tự `[11]`. (Cẩn thận vì 2.6 đổi sang hệ cơ số 2 không phải là `[11]` đâu, đừng có hiểu lầm)

### **Số viết theo khoa học, can, chỉ là gì?**

Thực ra, đây chỉ là những khái niệm rất cơ bản, nhưng khiến nhiều bạn bối rối.

- Thỉnh thoảng trong chương trình bạn gặp các số như `3.143E-6` (ý nghĩa là  $3.143 \times 10^{-6}$ ), hay số như `2.123E5` (ý nghĩa là  $2.123 \times 10^5$ ). Đó chính là các số viết theo kiểu khoa học. Qua 2 ví dụ trên chắc các bạn đã hiểu cách biểu diễn số thực theo dạng khoa học. (Bạn chỉ cần coi chữ E trong đó là 10 lũy thừa thôi mà!)

Khi bạn giải phương trình, thường gặp những sai số khá lằng nhằng kiểu như `6.141242342E-14`

Đừng có hoảng, vì nếu bạn đã hiểu cách viết số kiểu khoa học, bạn sẽ thấy những sai số đó rất nhỏ.

Chương trình có cung cấp cho bạn công cụ đổi số viết dạng khoa học sang số bình thường, đó là nút công cụ có hình kính lúp trên dãy công cụ ở phía trên cùng cửa sổ chính. Bạn hãy thử bật nó lên rồi lần lượt nhập vào vài số, ví dụ như: `6.141242342E-`

14; 12.345E3; 12.23456E-14 vùn vùn. Bấm ENTER sau mỗi lần nhập. Bạn sẽ thu được dạng số bình thường của các số viết theo khuôn dạng khoa học trên.

- Mỗi ngày, tháng, năm trong hệ thống âm lịch Việt Nam đều có tên, gọi là can, chi của ngày, tháng, năm đó. VD: năm 1984 của dương lịch ứng với năm Giáp Tý của âm lịch, vậy ta gọi Giáp là can của năm 1984, còn Tý là chi của năm 1984.

Can, chi của tháng 2 năm 1984 là Đinh Sửu. Đinh gọi là can của tháng 2 năm 1984, Sửu là chi của tháng 2 năm 1984. Tôi sinh vào tháng 2 năm 1984, theo lịch âm, tôi sẽ sinh vào tháng Đinh Sửu, năm Giáp Tý.

Có 9 can: Giáp, Ất, Bính, Đinh, Mậu, Kỷ, Canh, Tân, Nhâm, Quý.

Có 12 chi: Tý, Sửu, Dần, Mão, Thìn, Tị, Ngọ, Mùi, Thân, Dậu, Tuất, Hợi

Chương trình này có thể tìm can chi của tháng, của năm. Tạm thời tôi chưa tìm được tài liệu về việc tìm can chi của ngày, giờ, các bạn thông cảm.

## Các kiểu toán hạng

### *Toán hạng kiểu Số:*

Là các số thực, số nguyên thông thường VD số 5, số 3.14 là các toán hạng kiểu số.

### *Toán hạng kiểu Logic:*

T (True), F (False) là các toán hạng kiểu Logic. Hãy thử vài biểu thức sau và xem kết quả.

T and F or T->F

3 > 4 <-> 1 <= 1

T xor F -> (2+3)<=6

### *Toán hạng kiểu Ngày/giờ - toán hạng kiểu Chuỗi:*

- Trước hết hãy thử làm vài phép tính:

totime([2:15]+ hour(100))

[2:15:00]

[30/12/2002] + 100

[Số ngày tuổi của Hiệp là ] + (cdate - [4/2/1984])

- Những gì được viết trong cặp dấu ngoặc vuông, tùy nội dung, có thể được coi là toán hạng kiểu Ngày/giờ, cũng có thể được coi là toán hạng kiểu Chuỗi.

VD: [4/2/1984] hay [13:5:40 AM] hay [20/11/2002 13:50 PM] đều được coi là các toán hạng kiểu Ngày/giờ.

[Xin chào] hay [3.1416] hay [14/2] là các toán hạng kiểu Chuỗi.

- Nói chung, chương trình sẽ cố gắng đổi nội dung trong cặp dấu ngoặc vuông sang kiểu Ngày/giờ, nếu không đổi được thì sẽ coi đó là toán hạng kiểu Chuỗi.  
VD: [12/2002] không thể đổi sang kiểu Ngày/giờ nên bị coi là toán hạng kiểu Chuỗi
- Khi một toán hạng đã được chương trình coi là kiểu Ngày/giờ rồi thì cũng có thể coi toán hạng đó là kiểu Chuỗi hoặc kiểu Số thoải mái (Nói cách khác kiểu ngày giờ là kiểu lai căng giữa kiểu chuỗi và kiểu số, muốn coi nó là gì cũng được). Sau đây sẽ giải thích việc tại sao lại như vậy, một toán hạng kiểu Ngày/giờ nếu coi là toán hạng kiểu Số thì sẽ có giá trị bằng bao nhiêu, nếu coi nó là Chuỗi thì chuỗi sẽ như thế nào. Tuy nhiên, bạn chỉ cần quan tâm đến lợi ích mà toán hạng kiểu ngày/giờ mang lại chứ không cần biết đến bản chất của nó, vì vậy bạn có thể bỏ qua phần này.

Hãy xem qua bảng sau:

Toán hạng được viết là:	Nếu coi là kiểu Chuỗi	Nếu coi là kiểu Ngày/giờ	Nếu coi là kiểu Số
[30/12/1899]	[30/12/1899]	[30/12/1899 12:00:00 AM]	0
[31/12/1899]	[31/12/1899]	[31/12/1899 12:00:00 AM]	1
[4/2/1984]	[4/2/1984]	[4/2/1984 12:00:00 AM]	30716
[3:10:12]	[3:10:12]	[30/12/1899 3:10:12 AM]	0.13208333
[4/2/1984 3:10:12 AM]	[4/2/1984 3:10:12 AM]	[4/2/1984 3:10:12 AM]	30716.13208333

Nếu coi kiểu Ngày/giờ là một số thực có giá trị x thì:

- Phần nguyên của x là số thứ tự của ngày nếu lấy ngày 30/12/1899 là ngày thứ 0.

- Phần thập phân của x là giờ được đổi về dạng số.

Cách đổi giờ về dạng số như sau:

[0:0:1] tương đương với số  $1/(24*3600)$

[0:0:2] tương đương với số  $2/(24*3600)$

Tổng quát:

[h:m:s] tương đương với số  $((h*3600)+(m*60)+s)/(24*3600)$

VD: đổi [4/2/1984 3:10:12] về dạng số thực:

Ngày 4/2/1984 có số thứ tự là 30716 so với ngày 30/12/1899

3 giờ 10 phút 12 giây đổi sang kiểu số là:

$((3*3600)+(10*60)+12)/(24*3600) = 0.13208333$

Vậy [4/2/1984 3:10:12] tương đương với số 30716.13208333

Bạn không phải cố hiểu kĩ những thứ trên, tôi chỉ viết ra để bạn đỡ thắc mắc rằng tại sao một kiểu thời gian lại có thể coi là một con số được. Để thay cho việc đổi lằng nhằng trên chỉ cần dùng hàm **tonum** là xong. Hãy thử vài biểu thức sau:

**tonum([4/2/1984])**

**tonum([3:10:12])**

**tonum([4/2/1984 3:10:12])**

Hãy thử vài biểu thức cho kết quả là kiểu Chuỗi.

**canchi\_y(2002) dow([20/11/2002]) ptnt(2020)**

Hãy thử vài hàm phải dùng kiểu Chuỗi làm tham số:

Đây là vài hàm chuyển các số trong các hệ cơ số khác nhau về cơ số 10.

bin([10011]) hexa([1A0F]) base(16,[1A0F])

## Toán tử dành cho toán hạng kiểu số

### Các phép tính cơ bản:

Cộng  $+$ , trừ  $-$ , nhân  $*$ , chia  $/$ , giai thừa  $!$ , lũy thừa  $^$

Phép lấy phần nguyên của thương số trong phép chia 2 số nguyên: `div`, VD: `5 div 3`

Phép lấy phần dư trong phép chia 2 số nguyên: `mod`, VD: `10 mod 3`

**Lưu ý:** Phép tính lũy thừa:  $x^y$  chạy rất tốt nếu  $x$  (cơ số) dương. Nếu  $x$  âm thì phép tính chỉ thực hiện được khi  $y$  (số mũ) là số nguyên hoặc khi  $y$  có dạng là  $1/n$  (với  $n$  là số nguyên). Nguyên nhân vì khi cơ số âm, việc thực hiện lũy thừa một số âm với một số mũ thực là không tưởng tượng được.

VD:  $(-3)^{0.75} = -3^{3/4} = \sqrt[4]{(-3)^3}$  là phép tính không khả thi.

Hãy thực hành thử phép tính lũy thừa với số mũ âm hợp lệ

VD:  $-16^{(-1/4)} - (2^{-2})$

### Các hàm toán học cơ bản:

`sin`, `cos`, `tg`, `cotg`, `arcsin`, `arccos`, `arctg`, `arccotg` dùng với đơn vị là radian.

VD: muốn tính  $\sin(60^\circ)$  phải viết là `sin(pi/6)`.

Nếu thêm chữ `h` vào tên các hàm trên ta sẽ thu được các hàm lượng giác hypebolic.

VD1: tính sin hypebolic của 10 radian viết là `sinh(10)`

VD2: tìm xem góc nào có cos hypebolic là 1.5 viết là `arccosh(1.5)` (góc tìm được có đơn vị là radian)

Để đổi đơn vị góc từ độ sang radian dùng hàm `dtor` (degree to radian).

Để đổi ngược lại, dùng hàm `rtod` (radian to degree)

VD: muốn tính  $\sin(60^\circ)$  có thể viết là `sin( dtor(60) )`.

Ngoài ra còn có các hàm đổi qua lại giữa đơn vị: radian, degree, gradian, cycle, chỉ cần thay từ đầu và từ cuối trong tên hàm thành viết tắt của đơn vị cần đổi.

VD: muốn đổi 10 gradian sang cycle, viết là `gtoc(10)`, hay muốn đổi 10 degree sang cycle viết là `dtoc(10)` v.v...

Hàm tính lũy thừa số  $e$ : `exp` VD: để tính  $e^3$  viết là `exp(3)` (cũng có thể viết là `e^3`).

Các hàm `ln`, `log`, `lg`: VD: `ln10=ln(10)`; `log2=log(2)`; `lg28=lg(2,8)`

Hàm lấy trị tuyệt đối `abs`. VD: `|-10.3| = abs(-10.3) = 10.3`

Hàm lấy bình phương `sqr`; hàm lấy căn `sqr`. VD:  $(5.1)^2 = \text{sqr}(5.1)$ ;  $\sqrt{3.14} = \text{sqr}(3.14)$

Hàm tính tổ hợp `crn` và chỉnh hợp `arn`: VD: tổ hợp chập 3 của 10 phần tử = `crn(3,10)`

Hàm tìm ước chung lớn nhất và bội chung nhỏ nhất của 2 số nguyên: `ucln`, `bcnn`.  
VD: `ucln(27,81)`; `bcnn(27,81)`

### **Các hàm mở rộng:**

Hàm lấy trần của một số thực: `ceilling`. VD: `ceilling(3.1)` = 4

Hàm lấy sàn của một số thực: `floor` hoặc `trunc`. VD: `floor(3.7)` = 3

Hàm làm tròn số: `round`. VD: `round(3.7)` = 4

Hàm tạo số nguyên một cách ngẫu nhiên: `random`.

VD: tạo số nguyên ngẫu nhiên trong khoảng [7..100] viết là `random(7,100)`

Hàm chuyển các hệ cơ số từ 2->36 về hệ cơ số 10: `base`.

Mách nhỏ: Hàm `base` chỉ chuyển các số trong hệ cơ số từ 2->36 về hệ cơ số 10 mà không thể chuyển các số trong hệ cơ số khác cao hơn như hệ cơ số 50, 60... Lý do rất đơn giản: để biểu diễn các số trong hệ cơ số 16 cần 16 kí tự phân biệt (0..9,A..F), tương tự để biểu diễn các số trong hệ cơ số k cần k kí tự phân biệt. Ta chỉ có 36 kí tự phân biệt là 0..9,A..Z vì vậy không biểu diễn được số ở hệ cơ số cao hơn 36, trừ phi bạn bịa thêm kí tự.

VD1: chuyển số 1312 trong hệ cơ số 4 về cơ số 10 thì viết là: `base(4,[1312])`. Chú ý 1312 phải là 1 chuỗi nên phải trong ngoặc vuông.

VD2: chuyển số 1001 trong hệ cơ số 2 về cơ số 10 thì viết là: `base(2,[1001])`.

Hàm chuyển các cơ số 2, 8, 16 về cơ số 10: `bin`, `oct`, `hex`

VD1: chuyển số 1011 trong cơ số 2 về cơ số 10 viết là `bin([1011])` hoặc `base(2,[1011])`

VD2: chuyển số 1237 trong cơ số 8 về cơ số 10 viết là `oct([1237])` hoặc `base(8,[1237])`

VD3: chuyển số F0A trong cơ số 16 về cơ số 10 viết là `hex([F0A])` hoặc `base(16,[F0A])`

Hàm phân tích một số nguyên ra thừa số nguyên tố: `ptnt` (tạo ra một kiểu Chuỗi để thông báo kết quả).

VD `ptnt(1025)`

Hàm lấy số lớn/nhỏ hơn trong 2 số thực: `above` và `below`

VD: `above(3.5,2)` = 3.5; `below(4.12,4.3)` = 4.12

Hàm chuyển đổi sang kiểu số `tonum`:

Chuyển từ toán hạng kiểu Chuỗi sang kiểu số: không phải chuỗi nào cũng có thể chuyển được. Chỉ chuyển được vài chuỗi đặc biệt. VD nếu như chuỗi có dạng can, chi của năm, thứ trong tuần thì sẽ trả lại số thứ tự của chuỗi, nếu chuỗi có dạng số thì trả lại giá trị của số.

VD:

`tonum([Monday])`, `tonum(Tuesday)`

tonum([January]), tonum([February])  
tonum([Giáp]), tonum([Ât]) (chú ý: Ât không có dấu sắc)  
tonum([Tí]), tonum([Sửu])  
tonum([3.1416]), tonum([120.5]), <- chuỗi có dạng số  
tonum([4/2/1984 20:4:3])

### **Các toán tử thao tác bit:**

Các phép toán Logic, đều có thể dùng với các toán hạng kiểu Số (và phải là số nguyên). VD các biểu thức sau là hợp lệ và cho kết quả kiểu số.

3 xor 7

(5 and 2) or (not 10)

Nếu cố ý sử dụng với các số thực thì chương trình sẽ tự động làm tròn số trước khi thực hiện phép tính (điều này đã được thống nhất ở mục quy ước)

VD: 3.23 xor 6.712 cho kết quả tương đương với 3 xor 7.

Bản chất của các phép toán trên là việc thao tác trên các bit khi đổi các toán hạng sang hệ nhị phân.

VD1:

trong hệ nhị phân:

1001 or 1010 = 1011

1001 đổi sang hệ thập phân là 5

1010 đổi sang hệ thập phân là 10

1011 đổi sang hệ thập phân là 15

vậy 5 or 10 = 15

VD2: Bạn hãy thử thực phép tính sau:

bin([1010]) and bin([1001])

Sau đó nhấn Alt+2 để xem kết quả trong hệ nhị phân, sẽ thấy kết quả là [1000]

vậy thì trong hệ cơ số 2: 1010 and 1001 = 1000

## **Tính toán với kiểu Ngày/giờ**

Các ví dụ trong mục này đơn giản nhưng rất thú vị và hữu ích đấy!

- Hằng số lấy ngày giờ hiện tại cdate, ctime, cdatetime

Hãy thử vài ví dụ

VD1: cdate, ctime, cdatetime

VD2: Bao nhiêu ngày đã trôi qua kể từ ngày thành lập đảng? cdate-[3/2/1945]

VD3: Bây giờ tác giả bao nhiêu tuổi? (cdate-[4/2/1984])/365

VD4: 100 ngày nữa là ngày nào cdate+100

Lưu ý: Các hằng cdate, ctime và cdate time là các hằng đặc biệt, không được khai báo trong bảng quản lý hằng số. Các hằng này lấy ngày giờ trong hệ thống làm ngày giờ hiện tại. Vì vậy khi sử dụng các hằng này phải bảo đảm ngày giờ hệ thống chính xác.

- Hàm lấy thứ trong tuần: `dow` (day of week):

VD: hôm nay là thứ mấy? `dow(cdate)`; ngày 1/1/2010 là thứ mấy? `dow([1/1/2010])`

- Các hàm `hour` (giờ), `min` (minute: phút), `sec` (second: giây).

VD:

1024 giây nữa là thời điểm nào? `cdatetime+sec(1024)`

1000 phút nữa là thời điểm nào? `cdatetime+min(1000)`

$2^{10}$  giờ nữa là thời điểm nào? `cdatetime+hour(2^10)`

Bạn có thể thay thế hằng số `cdatetime` trên thành bất kì một toán hạng kiểu Ngày/giờ khác (hoặc các biểu thức cho kết quả là kiểu Ngày/giờ).

- Hàm `totime`, `todate`, `todatetime`:

Bây giờ hãy thử biểu thức sau: `[10:2:45]+sec(100)`

Bạn sẽ thấy kết quả là `[30/12/1899 10:04:25 AM]`. Vậy 30/12/1899 ở đâu ra?

Một toán hạng kiểu Ngày/giờ chính xác thì phải đủ 2 thành phần: ngày và giờ. Nếu thiếu một trong hai thành phần ngày hoặc giờ thì nó sẽ lấy ngày và giờ mặc định thế vào chỗ thiếu. Ngày mặc định là 30/12/1899, giờ mặc định là 12:0:0 AM. Vậy nên khi bạn viết toán hạng `[10:2:45]` thì thực sự chương trình sẽ tự bổ sung thêm phần ngày bị thiếu trở thành `[30/12/1899 10:2:45 AM]` trước khi tính toán.

Vậy thì biểu thức được thực hiện sẽ là `[30/12/1899 10:2:45 AM] + sec(100)` và cho kết quả như trên.

Nếu bạn thấy khó chịu với cái đuôi ngày được thêm vào tự động và chỉ muốn xuất riêng thành phần giờ ra thôi thì bổ sung thêm hàm `totime` vào biểu thức thành: `totime([10:2:45]+sec(100))`

Tương tự, nếu chỉ muốn xuất phần ngày của một toán hạng kiểu Ngày/giờ thì dùng hàm `todate`

Hàm `todate`, `totime`, `todatetime` còn có công dụng lớn hơn nữa là có thể chuyển đổi một kiểu bất kì sang kiểu Ngày/giờ.

VD: chuyển số -10.12 sang kiểu Ngày/giờ viết là: `todatetime(-10.12)`.

Sau khi làm xong VD trên, hãy ghi nhớ kết quả. Bây giờ hãy gõ biểu thức này vào: `-10.2` (gõ ENTER). Sau đó tích vào mục **Đổi sang kiểu Ngày/giờ**. Hãy so sánh kết quả thu được lúc này với kết quả trên. Thấy giống hệt nhau phải không.

Vậy công dụng của mục chọn **Đổi sang kiểu Ngày/giờ** khi được đánh dấu chính là đổi tất cả các kết quả tính toán ra kiểu Ngày/giờ, trừ khi kết quả tính toán có kiểu Logic (không thể đổi ra kiểu Ngày/giờ được).

Còn công dụng của các mục chọn Bin, Dec, Hex, Other (phía trái mục) chọn này là để xuất kết quả dưới dạng chuỗi sau khi đổi kết quả về hệ cơ số được chỉ định (chắc bạn cũng đã đoán ra, nên tôi không giải thích thêm).

- Tính ngày công nguyên: hàm `countday`:

VD1: Đếm xem bao nhiêu ngày đã trôi qua kể từ ngày đầu tiên của năm công nguyên (ngày 1/1/0000) cho tới bây giờ.

`countday(cdate)`



VD2: Đếm xem bao nhiêu ngày đã trôi qua kể từ ngày đầu tiên của năm công nguyên (1/1/0000) cho tới lúc tác giả sinh ra.

`countday([4/2/1984])`

- Hàm ngược của hàm `countday`: `eraday`

VD: Tính xem ngày thứ 1 triệu kể từ ngày đầu tiên của công nguyên là ngày nào?

`eraday(10^6)`

- Các hàm tính can, chi của năm: `can_y`, `chi_y`, `canchi_y`. (đuôi y trong tên hàm là viết tắt của year: năm)

VD: muốn tính can, chi của năm 1984 (năm tuổi của tác giả), thì viết là:

`canchi_y(1984)` hoặc là `canchi_y([4/2/1984])`.

ở cách viết thứ 2, chương trình chỉ để ý tới phần năm của tham số `[4/2/1984]` (kiểu ngày), còn ngày và tháng có thể điền bừa vào (tất nhiên là phải điền hợp lý).

- Tương tự như vậy, ta cũng có các hàm tìm can, chi của tháng với đuôi là m (`month`): `can_m`, `chi_m`, `canchi_m`.

VD: tìm can chi của tháng 1 năm 2003 viết là `canchi_m([1/1/2003])`.

Trong biểu thức trên, chương trình chỉ để ý tới thành phần tháng và năm của tham số `[1/1/2003]` (kiểu ngày). Còn thành phần ngày, cứ điền bừa vào cho đủ (tất nhiên là phải hợp lý. VD như điền là 100/1/2003 là không được!)

**Lưu ý:** Khi dùng các hàm can chi. VD: muốn tìm can chi của tháng 2 năm 1984 thì có thể viết là:

`canchi_m([4/2/1984])`.

Các bạn thường hay quên dấu mở đóng ngoặc vuông, thành ra `canchi_m(4/2/1984)` nếu vậy, đầu tiên chương trình sẽ thực hiện các phép chia:

$4/2/1984 = 0.00100806451612903$

Sau đó tìm `canchi_m(0.00100806451612903)`. Con số dài đặc này sẽ được tự động đổi ra kiểu Ngày/giờ (vì tham số của hàm `canchi_m` phải là kiểu Ngày/giờ). Khi đổi sang kiểu Ngày/giờ, nó là `[30/12/1899 12:01:27 AM]`, cuối cùng kết quả sẽ là can chi của tháng 12 năm 1899 trong khi đang mong đợi can chi của tháng 2 năm 1984.

Vì vậy, khi tìm can chi của một số thực, chương trình thường cảnh báo bạn cẩn thận. Thiếu cặp dấu ngoặc vuông. Nếu bạn chắc chắn thì không cần quan tâm đến lời cảnh báo đó.

## Toán tử dành cho toán hạng kiểu Logic

Chỉ có vài toán tử loại này, rất đơn giản.

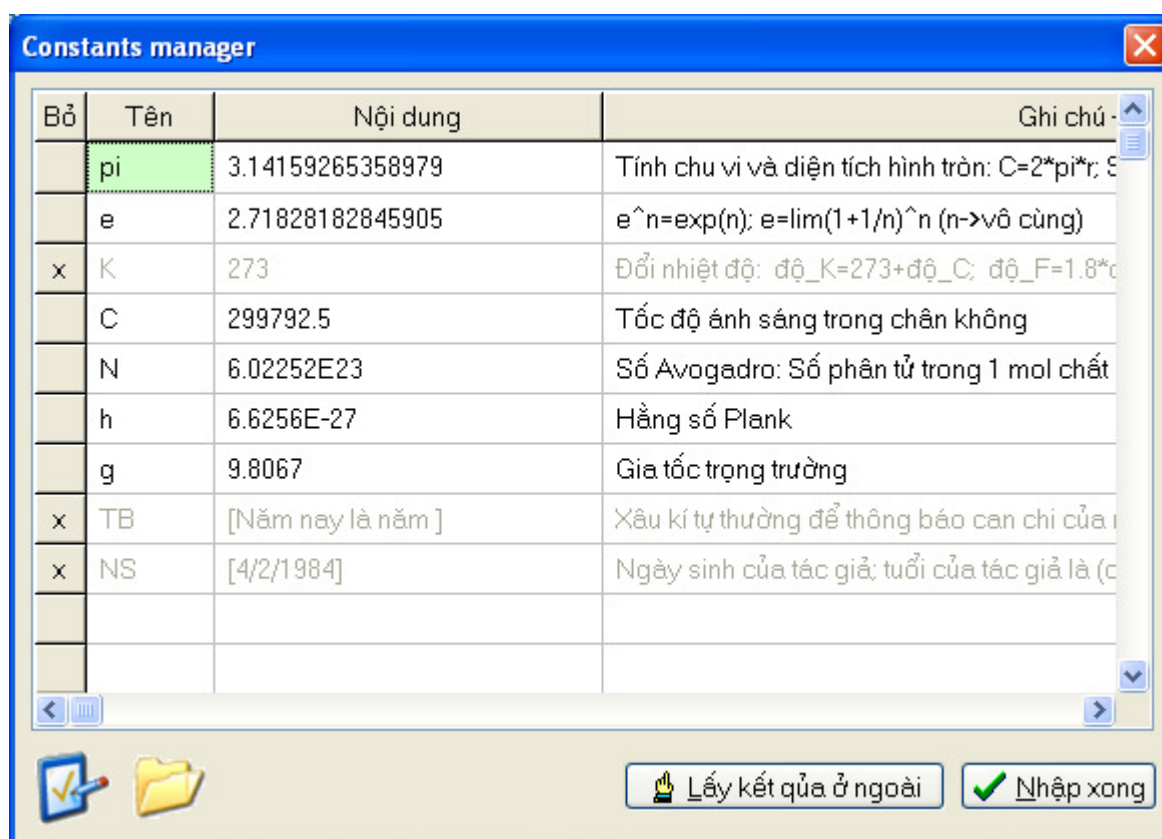
- Phép hội: `and`
  - Phép tuyển: `or`
  - Phép phủ định: `not`
  - Phép kéo theo: `->`
  - Phép tương đương: `<->`
  - Phép `xor`: gọi p, q là các biểu thức Logic, p `xor` q cho `F` khi hai biểu thức p, q có cùng giá trị chân lý.
  - Các phép so sánh 2 toán hạng kiểu số bao gồm: `=`, `>`, `<`, `>=`, `<=`, dùng để so sánh 2 toán tử với nhau cho kết quả là một toán hạng mới kiểu Logic.
- VD: `3<=2` cho kết quả là `F`

**(2+7)=10** cho kết quả là T.

- Vì thời gian làm chương trình có hạn nên tôi chưa kịp bổ sung thêm tiện ích lập bảng giá trị chân lí của biểu thức Logic bất kì. Hẹn tới phiên bản sau của chương trình.

## Quản lý hằng số:

Bấm vào nút công cụ có biểu tượng chữ C trên thanh công cụ, thấy hộp thoại Constants Manager hiện ra:



Trên đây là những hằng số mặc định, thông tin của các hằng số này được lưu trong tệp DEFAULT.CST. Nếu không có tệp tin này thì sẽ không có gì trong bảng hằng số cả. Tất cả các hằng số trong chương trình, trừ các hằng số đặc biệt cdate, ctime, cdatetime ra, đều được lưu giữ tại đây.

Để sửa một thông tin ở ô nào đó thì đưa hộp xanh tới đó ấn F2 hoặc ENTER, sửa xong, lại ấn ENTER để ghi nhận. Muốn di chuyển một dòng nào đó đi thì click trái chuột vào cột đầu tiên dòng đó và kéo đến vị trí mong muốn.

Để khoá hoặc mở khoá lại một hằng số, thì click phải chuột vào cột đầu tiên (cột Bỏ) của dòng chứa hằng số đó. Một hằng số bị khoá thì sẽ chỉ có thể làm cảnh trong bảng quản lý hằng số mà không thể sử dụng được trong chương trình.

Hãy thử khoá hằng số tên là pi lại (click trái chuột vào ô ở cột đầu dòng đầu để hiện ra dấu gạch chéo ở đó). Nhấn nút Nhập xong. Bây giờ, trong cửa sổ chính hãy tính thử biểu thức:  $\sin(\pi/2)$ , sẽ bị báo lỗi, vì lúc này coi như hằng số pi đã bị khoá rồi, coi như không tồn tại trong bảng quản lý hằng. Hãy mở lại bảng quản lý hằng số và mở khoá cho hằng pi thì lại sử dụng được bình thường.

Khi sử dụng hằng số thì phải lưu ý, viết tên hằng chính xác cả về việc viết chữ hoa hay chữ thường. VD: bạn khai báo một hằng tên là pi (toàn bộ là chữ thường) trong bảng quản lý hằng. Trong cửa sổ chính, bạn viết biểu thức là  $\sin(\pi/2)$  (viết hoa chữ P), chương trình sẽ báo lỗi, Nó chỉ biết hằng số pi (toàn chữ thường), chứ không biết hằng số Pi (P hoa), vì hằng số Pi (P hoa) không được khai báo trong bảng quản lý hằng.

Hai nút ở góc trái dưới bảng quản lý hằng số có tác dụng mở và lưu trữ thông tin trong bảng vào tập tin đuôi CST. Các bạn có thể dùng chương trình soạn thảo văn bản bất kì nào đó để đọc tập tin dạng này, việc sửa chữa trực tiếp những tập tin này cũng rất đơn giản.

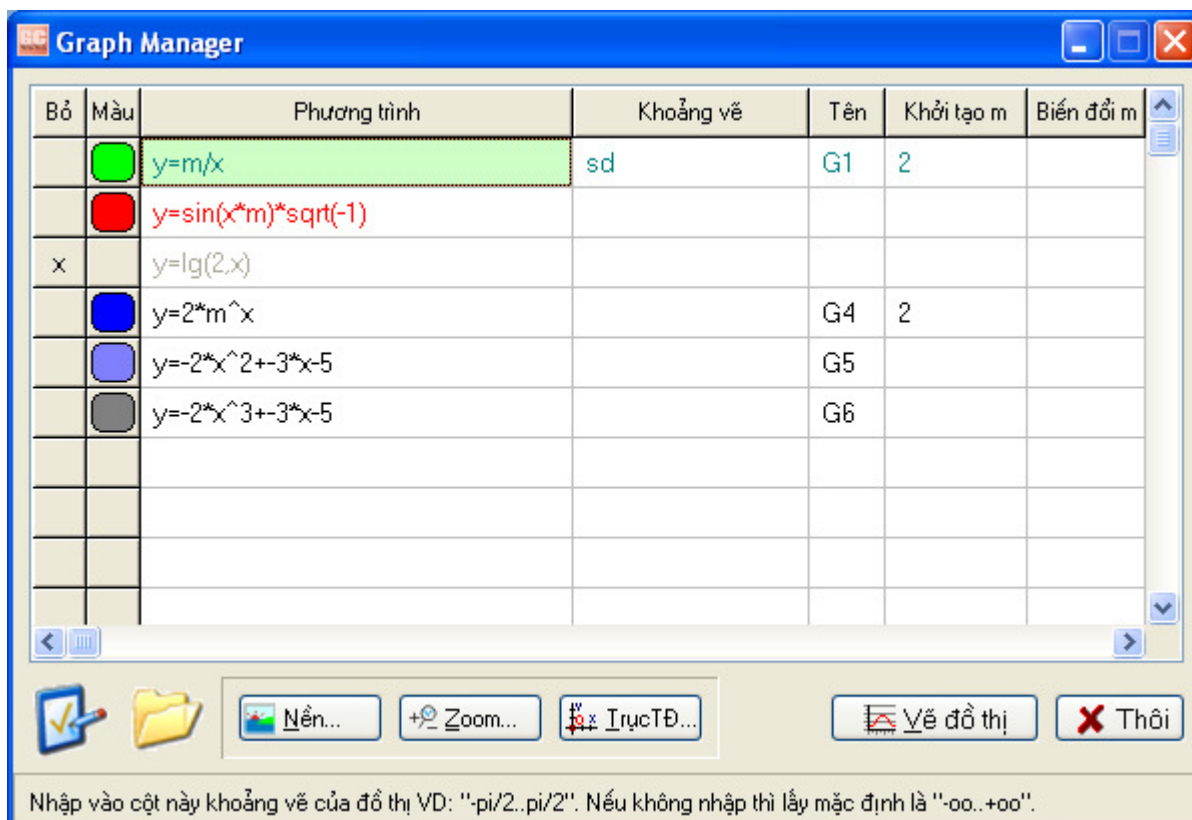
Nút Lấy kết quả ở ngoài sẽ bổ sung vào bảng một hằng số có giá trị là kết quả của biểu thức ở cửa sổ chính, có tên do chương trình tự đặt. Dĩ nhiên, các bạn có thể tuỳ ý sửa các thông tin này.

Còn khi bạn nhập một hằng số mới, thì chỉ cần nhập tên và nội dung của nó vào bảng hằng số mà thôi, còn cột ghi chú ý nghĩa, bạn thích thì nhập, không nhập cũng được.

## Về đồ thị

### *Trong cửa sổ quản lý đồ thị:*

Bấm vào nút biểu tượng đồ thị trên thanh công cụ, thấy một hộp thoại hiện ra. Các thao tác trong hộp thoại này giống như thao tác trong bảng quản lý hằng số ở trên. Trước hết hãy Click vào nút mở (màu vàng, to đùng ở góc trái dưới hộp thoại). Mở tệp EXAMPLE.GR ra. Nếu không có tập tin này thì thử nhập vài đồ thị vào bảng, kiểu như sau: (Đừng quan tâm đến sự khác biệt về màu chữ giữa các dòng trong bảng)



thử Click phải chuột vào ô màu xanh lá cây (ô ở dòng 2 cột 2) để chọn lại màu cho đồ thị, Hãy thử chọn một màu rồi nhấn vào nút Vẽ đồ thị. Các đồ thị trên sẽ được vẽ vào cửa sổ vẽ.

### ***Trong cửa sổ vẽ:***

#### ***Di chuyển và thay đổi trực tiếp màu sắc đồ thị:***

- Nhấn SPACE để bật tắt cửa sổ Short Information (thông tin ngắn). Trong cửa sổ này, bạn có thể Click vào ô chứa thông tin về màu sắc đồ thị để thay đổi trực tiếp màu sắc đồ thị mà không phải quay lại cửa sổ quản lý đồ thị.
- Nhấn các **phím mũi tên** để di chuyển gốc tọa độ. Nếu giữ **CTRL** trong quá trình di chuyển đồ thị thì tốc độ di chuyển sẽ tăng gấp đôi. Tốc độ di chuyển, vị trí gốc tọa độ, màu sắc trục tọa độ được quyết định trong hộp thoại Origin Option. Có 2 cách mở hộp thoại này:  
 Cách 1: Nhấn phải chuột trong cửa sổ vẽ, chọn mục Trục/gốc tọa độ hoặc bấm phím **G** trên bàn phím  
 Cách 2: Nhấn nút Trục TĐ trong cửa sổ quản lý đồ thị.

#### ***Phóng lớn/thu nhỏ đồ thị:***

- Bấm phím **+**, và **-** (ở khu vực bàn phím mở rộng) để ZOOM (thu phóng) đồ thị.
- Giữ phím **SHIFT** và dùng các phím mũi tên để ZOOM đồ thị theo từng trục tọa độ (tức là co dãn đồ thị theo chiều ngang hoặc chiều dọc), nhớ nhìn sự thay đổi của các vạch chia đơn vị trên các trục tọa độ.

- Giá trị ZOOM càng lớn thì đồ thị càng được phóng to. Thông số về giá trị Zoom và độ biến đổi các giá trị này khi dùng các phím trên được quyết định trong hộp thoại Zoom Option. Có 2 cách mở hộp thoại này tương tự như 2 cách mở hộp thoại Origin Option ở trên.

Cách 1: Nhấn phải chuột trong cửa sổ vẽ, chọn mục Thu/phóng đồ thị hoặc bấm phím **Z** trên bàn phím

Cách 2: Nhấn nút **Zoom** trong cửa sổ quản lý đồ thị.

### Quan sát sự thay đổi của đồ thị khi thay đổi tham số

Nhấn phím **PGUP**, **PGDOWN** để tăng, giảm giá trị của các tham số (m). Độ tăng giảm các tham số này được khai báo trong cột biến đổi m của bảng quản lý đồ thị, nếu không khai báo thì lấy mặc định là 1. Tức là nhấn PGDOWN thì giá trị các tham số trong phương trình được tăng 1, Nhấn PGUP thì giảm 1. Quan sát sự thay đổi của đồ thị và giá trị tham số m tương ứng trong cửa sổ thông tin ngắn (dùng phím SPACE để bật tắt cửa sổ này).

### Quay lại cửa sổ quản lý đồ thị

Cách 1: Nhấn **F5** để chuyển đổi qua lại giữa cửa sổ vẽ và cửa sổ quản lý đồ thị.

Cách 2: Nhấn **ESC** để quay lại cửa sổ quản lý đồ thị đồng thời cập nhật thông tin.

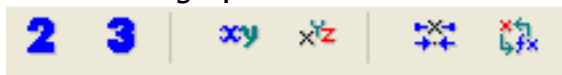
Khi đang ở cửa sổ vẽ, bạn có thể thay đổi giá trị tham số (nhấn PGUP, PGDOWN), thay đổi màu sắc đồ thị (nhấn vào ô chỉ màu của cửa sổ Short Information). Nếu bạn dùng ESC để quay lại cửa sổ quản lý thì mọi thay đổi trong cửa sổ vẽ (về màu sắc, giá trị tham số) sẽ được cập nhật vào cửa sổ quản lý, còn phím F5 chỉ đơn thuần chuyển đổi qua lại giữa 2 cửa sổ mà không thay đổi gì thông tin của 2 cửa sổ cả.

Còn chức năng xuất đồ thị thành ảnh \*.BMP và hộp hội thoại Background Option, bạn hãy tự tìm hiểu lấy. Rất đơn giản thôi.

## Sử dụng các công cụ giải phương trình

### **Tổng quan:**

Sáu nút công cụ đầu tiên trên thanh công cụ sẽ giúp bạn giải phương trình.



Cứ nhấn vào một nút thì sẽ có hộp thoại hiện ra, nói chung là giao diện rất dễ hiểu. Có 2 điểm thuận lợi mà bạn nên biết (Đây là điểm chung của tất cả các hộp thoại được gọi bởi nút công cụ)

- Có thể cùng lúc mở cả tất cả các hộp thoại. Dùng chuột liên hệ các hộp thoại này với nhau và với cửa sổ chính (và với cả cửa sổ vẽ đồ thị nữa). Việc này thuận tiện cho việc sao chép, cắt dán qua lại kết quả tính toán giữa các cửa sổ với nhau. Khi bạn đang ở bất cứ hộp thoại nào, có thể nhấn **F10** để trở về cửa sổ chính ngay lập tức (từ cửa sổ chính có thể gọi lại ra mọi cửa sổ khác).

- Hãy để ý thấy trong 6 hộp thoại phục vụ giải các phương trình bậc 2 cũng có ô văn bản màu tím nhạt, thử đánh chữ linh tinh vào ô đó, thấy chữ trong ô màu đỏ. Vậy ô đó để làm gì?

Khi nhập dữ liệu để giải phương trình nào đó, giả sử giải phương trình bậc 2 cần nhập 3 hệ số chẳng hạn. Thay vì nhập vào trực tiếp hệ số, bạn có thể nhập vào các biểu thức mà cho kết quả là các hệ số đó. Nếu biểu thức bạn nhập sai thì biểu thức sai sẽ được đưa vào ô màu tím nhạt này, còn thông báo lỗi sẽ thể chỗ của biểu thức bị lỗi. Bạn có thể sửa lại biểu thức ngay trong ô màu tím nhạt rồi Copy trả về chỗ cũ. Nếu thấy hơi khó hiểu hãy đọc ví dụ sau:

Giả sử bạn muốn giải phương trình bậc 2 sau:  $\sqrt{2\pi} \cdot x^2 + (\sqrt{3,9}+2) \cdot x \pi/2$

+) Bấm vào nút công cụ thứ nhất (biểu tượng số 2 màu tím) hoặc nhấn F2.

+) Nhập 3 hệ số như sau:  $\sqrt{2\pi}$ ,  $9^{(1/3)}+2$ ,  $-\pi/2$  rồi nhấn vào nút [Đã nhập Xong], kết quả sẽ hiện ra.

Bạn thấy đấy thay vì nhập hệ số là số cụ thể, tôi nhập các biểu thức vào các ô hệ số và vẫn ra kết quả ngon lành!

Bây giờ bạn hãy cố ý nhập sai hệ số a thử xem sao, thay vì nhập đúng là  $\sqrt{2\pi}$ , bạn hãy nhập thiếu dấu nhân thành  $\sqrt{2\pi}$ , các hệ số khác cứ giữ nguyên đó Rồi, bấm vào nút Đã nhập Xong xem Bạn thấy đấy, biểu thức lỗi đã được đưa vào ô màu tím nhạt, còn vị trí cũ của biểu thức lỗi bây giờ chứa thông báo lỗi. Bạn hãy sửa biểu thức lỗi trong ô màu tím nhạt rồi cắt và dán nó vào vị trí cũ.

### **Nút công cụ thứ 1 -> 4:**

Bốn nút công cụ đầu tiên lần lượt giúp bạn giải các phương trình bậc 2, bậc 3, các hệ phương trình bậc nhất 2 ẩn, 3 ẩn. Cách sử dụng rất đơn giản, cứ thử thì biết.

Mách nước: Với công cụ giải phương trình bậc 2, bạn còn có thể nhập nghiệm để tìm hệ số thay vì nhập hệ số rồi tìm nghiệm như bình thường. Để sử dụng tính năng này, bạn phải bảo đảm các ô hệ số được xoá sạch.

VD: Cần tìm phương trình bậc 2 có nghiệm là  $\sqrt{0.1}$  và  $\sqrt[3]{2}$  hãy làm như sau:

- Xoá tất cả nội dung trong các ô hệ số.

- Nhập vào 2 ô nghiệm lần lượt là  $\sqrt{0.1}$  (hoặc  $0.1^{(1/2)}$ ) và  $2^{(1/3)}$

- Nhấn vào nút [Đã nhập Xong] (hoặc nhấn Alt+X hoặc ENTER).

### **Nút công cụ thứ 5:**



Tìm nghiệm gần đúng của phương trình  $f(x)=0$ .

Tác dụng của nút này là ép khoảng nghiệm. Đầu tiên bạn cho biết nghiệm X cần tìm nằm trong một khoảng nào đó VD nằm trong khoảng từ  $a \rightarrow b$ , công cụ này sẽ giúp bạn thu hẹp dần khoảng cách a, b thậm chí đến khi a và b trùng với nghiệm X cần tìm luôn.

Nguyên tắc thu hẹp khoảng nghiệm rất đơn giản, dựa trên định lý cơ bản là nếu nghiệm x nằm trong khoảng  $[a,b]$  thì  $f(a) \cdot f(b) < 0$ .

đầu tiên, chương trình sẽ chia đôi khoảng  $[a,b]$ , điểm c là điểm giữa của a,b. Chương trình sẽ kiểm tra xem nếu  $f(a) \cdot f(c) < 0$  tức là nghiệm nằm trong khoảng  $[a,c]$ , nếu  $f(c) \cdot f(b) < 0$  thì nghiệm nằm trong khoảng  $[c,b]$ . Thế là xong một lần ép khoảng. Như vậy, kiểu gì thì sau

mỗi lần ép khoảng thì khoảng chứa nghiệm cũng hẹp lại còn phân nửa lúc trước đó. Số lần ép khoảng càng nhiều thì khoảng nghiệm càng hẹp, có khi hẹp tới nỗi trùng luôn vào nghiệm cần tìm. Bạn có thể xác định tính chính xác của nghiệm dựa vào số lần ép khoảng, hoặc dựa vào sai số cho phép, sai số này được tính bằng hiệu  $b-a$ .

**Lưu ý:** Chỉ tìm được nghiệm trong điều kiện sau:

- Phương trình có duy nhất một nghiệm trên khoảng  $[a,b]$
- Phương trình đơn điệu (đồng biến hoặc nghịch biến) trên khoảng  $[a,b]$

Xem ra thật phức tạp để khởi tạo được  $a, b$  thoả mãn những điều kiện trên, nhất là với những phương trình phức tạp, nhưng với công cụ vẽ đồ thị trong tay, việc xác định  $a, b$  rất dễ dàng.

VD:

Bạn muốn tìm nghiệm của phương trình  $2x^4 - 3x - 1 = 0$  (phương trình 1)

- Trước hết bạn hãy vẽ đồ thị hàm  $y=2x^4-3x-1$ . (xem phần vẽ đồ thị)
- Dựa vào đồ thị, bạn biết được phương trình có 2 nghiệm (đồ thị cắt trục hoành tại 2 điểm). Giả sử bạn muốn tìm nghiệm nhỏ.

Dựa vào đồ thị bạn biết được rằng nghiệm nhỏ này nằm trong khoảng từ -1 đến 0 (Bạn có thể lấy khoảng khác, miễn sao khoảng đó thoả mãn đủ điều kiện ở phần lưu ý trên là được), trong khoảng này thấy hàm số liên tục (đồ thị liền nét trong khoảng này) và đơn điệu (nghịch biến). Vậy là đủ điều kiện để thực hiện ép khoảng này.

- Trở về cửa sổ chính, hãy bật công cụ 5 lên,
  - + Nhập vào ô đầu tiên biểu thức  $2*x^4-3*x-1$
  - + Nhập vào khoảng nghiệm  $-1, 0$
  - + Chọn mục [Sai số], và gõ vào ô bên cạnh là  $10^{-20}$  hoặc  $1E-20$
  - + Nhấn vào nút [Bắt đầu tính]
  - + Đợi một lúc (1 giây thôi) thấy 2 mốc khoảng nghiệm đều bằng -0.3258 ( $a=b=-0.3258$ ) tức là khoảng nghiệm bị thu hẹp đến mức đã trùng với nghiệm rồi. Bạn có thể nhấn [Tạm dừng], vì không cần thiết phải ép khoảng thêm nữa. Vậy là nghiệm nhỏ của phương trình bằng -0.3258.

+ Bây giờ bạn hãy thử tìm tiếp nghiệm lớn xem. Dựa vào đồ thị, bạn thấy nghiệm lớn nằm trong khoảng 1..2, Bạn hãy nhấn nút [Reset], sau đó nhập lại khoảng nghiệm là 1..2, rồi nhấn [Bắt đầu tính]. Đợi một lúc thấy 2 mốc khoảng nghiệm đều bằng 1.2393 Bạn có thể nhấn tạm dừng. Vậy là nghiệm lớn của phương trình bằng 1.2393

### **Nút công cụ thứ 6:**



Tìm nghiệm gần đúng của phương trình  $x=f(x)$ . Sử dụng công cụ này không dễ.

Công cụ này chỉ sử dụng được khi  $f(x)$  là hàm hội tụ. Cách xác định một hàm có hội tụ hay không khá là rắc rối, tôi không muốn viết ra đây vậy nên sau khi tính xong, tốt nhất bạn nên thử lại kết quả. Cách tính nghiệm của công cụ này như sau:

$x_0$ =Nội dung trong ô [Khởi tạo nghiệm]

$x_n=f(x_{n-1})$

Tức là chương trình sẽ tìm nghiệm bằng cách tính  $x=f(f(f(f(x_0))))$



Với n đủ lớn (nếu hàm hội tụ thì thực sự n chỉ cần bằng 20, 30 lần là đủ) thì kết quả sẽ rất chính xác.

(Xem mục Nút nhấn X ở dưới để biết cách thử lại phương trình nhanh chóng)

VD1:

Muốn tính  $\sqrt{2}$  ta sẽ tìm nghiệm của phương trình  $x^2-2=0$

Biến đổi phương trình thành  $x=(1/2)(x+2/x)=f(x)$

Ta sẽ giải phương trình này như sau:

Nhập vào ô trên cùng biểu thức:  $(x+2/x)/2$ . Khởi tạo nghiệm bằng mấy cũng được, miễn sao thuộc tập xác định (hãy thử khởi tạo bằng 10 xem) cho số lần lặp là 1000 luôn! Nhấn nút Bắt đầu Lặp, (trong khi nhấn, hãy nhìn sẵn vào ô Nghiệm có màu sẫm) được một lúc (rất nhanh thôi), bạn thấy rằng nội dung ô Nghiệm là 1.4142135623731, việc lặp sẽ vẫn tiếp tục tiến hành, cho đến hết 1000 lần lặp thì thôi, nhưng lúc này, không có lặp nữa thì nghiệm vẫn như vậy thôi, bạn có thể nhấn tạm dừng.

VD2:

Bây giờ hãy thử giải phương trình trong VD ở phần **nút công cụ thứ 5**, phương trình đó là:

$$2*x^4-3*x-1=0$$

Biến đổi phương trình trên thành phương trình:

$$x=(2*x^4-1)/3$$

Đầu tiên bạn phải bấm Reset để nhấn của nút Tiếp tục biến thành nút Bắt đầu Lặp

Bây giờ nhập biểu thức  $(2*x^4-1)/3$  vào ô đầu tiên, Khởi tạo nghiệm là 0, cho số lần lặp là 100;

Nhấn nút [Bắt đầu Lặp] sẽ tìm được nghiệm là -0.325820205251213, đây chính là nghiệm lớn của phương trình.

Đừng vội mừng! Bây giờ hãy thử nhấn [Reset], rồi thay đổi Khởi tạo nghiệm sao cho khác 0 (khởi tạo lúc này là 0) rồi bấm [Bắt đầu Lặp] với khởi tạo mới này xem, vừa lặp đến lần thứ 2 thì đã bị lỗi rồi! Sự bất ổn định của công cụ này là thế đấy!

**Lưu ý:** Cứ tới mỗi một lượt tính mới, bạn phải nhấn [Reset] để xoá bỏ những ảnh hưởng của lượt tính cũ.

**Lời khuyên:** Vì tôi làm chương trình này trong thời gian khá gấp gáp, nên cũng chưa đầu tư nhiều cho nút công cụ thứ 6 này. Thực sự thì những hiểu biết của tôi về vấn đề dùng phương pháp lặp tìm nghiệm gần đúng của phương trình một ẩn chưa được sâu sắc lắm, nút công cụ này lại được xây dựng vội vã dựa trên cơ sở của vấn đề đó. Tôi cứ làm tạm một công cụ như vậy, với hi vọng bạn sẽ giúp tôi nghĩ ra cách ứng dụng có ích cho công cụ này (bởi tôi vẫn không tin là nó vô dụng!). Thực ra, bạn kết hợp tiện ích vẽ đồ thị và nút công cụ thứ 5 là bạn đã có trong tay một phương tiện khá hoàn hảo để giải tất cả các phương trình một ẩn của toán sơ cấp rồi!

### **Nút công cụ hình kính lúp**

Như ta đã biết, định nghĩa số hữu tỉ là số có thể được biểu diễn dưới dạng phân số  $a/b$ , trong đó,  $a, b$  là các số nguyên, công cụ này giúp tìm ra  $a, b$  của một số hữu tỉ bất kì. Hãy bật nó lên và thử vài ví dụ sau:

VD1: Muốn đổi số 2.10 234 234 234... sang phân số:



Đây là số hữu tỉ vô hạn tuần hoàn với chu kì là 234.

Ta nhập vào là 2.10(234) rồi nhấn ENTER

Bạn sẽ nhận được kết quả là phân số **5834:2775**.

Nếu muốn thử lại kết quả đó, hãy copy kết quả vào ô Nhập biểu thức, sửa dấu hai chấm của phân số thành dấu sổ chéo của phép chia, biểu thức sẽ là **5834/2775**, giờ thì nhấn ENTER, thấy chưa! kết quả là số hữu tỉ 2.10 234 234 234...

VD2: Muốn đổi số 3.3333... sang phân số:

Đây là số hữu tỉ vô hạn tuần hoàn với chu kì của là 3.

Ta nhập vào là 3.(3) rồi nhấn ENTER

VD3: Đổi 3.14 sang phân số:

Số hữu tỉ này không vô hạn, không có chu kì.

Ta nhập vào là 3.14 rồi nhấn ENTER

Ngoài tác dụng tìm dạng phân số, nút công cụ này còn có một tác dụng mở rộng nữa! Đó là tìm dạng bình thường của số viết theo khuôn dạng khoa học, hãy nhập vào các số viết theo dạng khoa học ví dụ như: **6.141242342E-14**; **12.345E3**; **12.23456E-14** vân vân. Bấm ENTER sau mỗi lần nhập. Bạn sẽ thu được dạng số bình thường của các số viết theo khuôn dạng khoa học trên.

## ***Nút nhấn X***

Đó là nút nằm cạnh hộp nhập biểu thức có nhãn là chữ **X** (có thể nhấn ALT+X để kích hoạt nó).

Xét biểu thức:  $2x^4 + 3x^2 + 2x + 1$

Giả sử bạn muốn tính biểu thức trên với giá trị x lần lượt là:  $x=\pi/2$ ;  $x=\pi/4$ ;  $x=\sin(\pi^2)$ , Bình thường bạn phải thay lần lượt 3 giá trị x này vào phương trình để tính, tức là phải viết đi viết lại 3 lần. Nhưng mọi việc sẽ đơn giản hơn khi dùng công cụ này. Hãy làm như sau:


Đầu tiên, hãy viết nguyên xi biểu thức này vào ô Biểu thức cần tính ở cửa sổ chính:

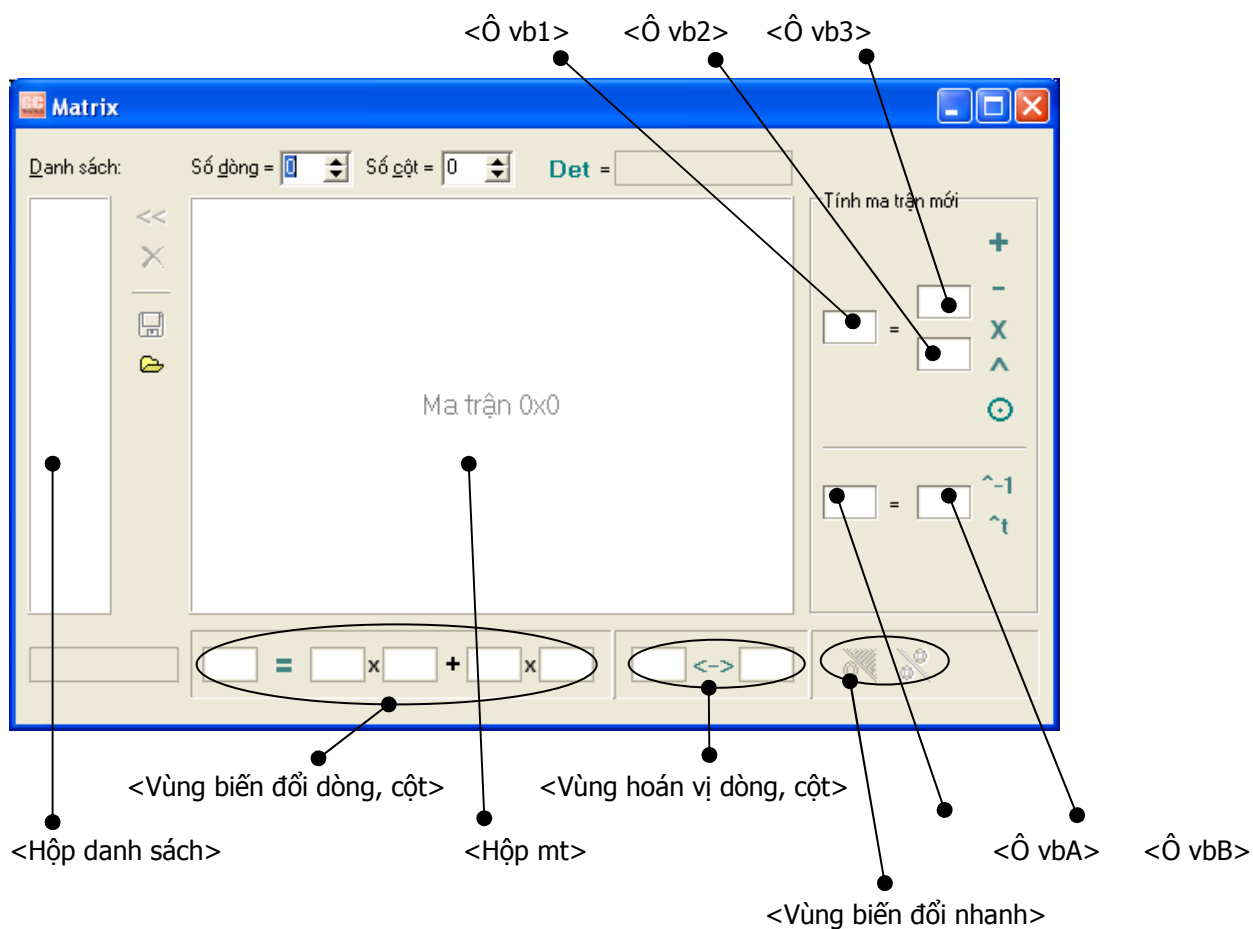
**$2*x^4+3*x^2+2*x+1$**

Nhấn vào nút công cụ có nhãn là chữ **X** sau đó nhập vào là  $\pi/2$  rồi nhấn [Ok] vậy là bạn đã tính được giá trị biểu thức với  $x=\pi/2$  rồi! các giá trị còn lại làm tương tự.

Nút này dùng rất tốt khi muốn thử lại nghiệm của phương trình một lần.

## **Ma trận**

Bấm vào nút có biểu tượng các đường kẻ caro màu tím  trên thanh công cụ. Hộp thoại Matrix hiện ra (xem hình và hãy thuộc tên các đối tượng trên hộp thoại này):



### ***Tạo và hiệu chỉnh ma trận***

Đặt kích thước ma trận vào 2 ô văn bản (ô vb) trên cùng, bên trái, nhấn ENTER.

Lưu ý: Khi muốn cập nhật lại kích thước ma trận theo thông số khai báo trong 2 ô vb đó phải đặt con trỏ nhấp nháy ở một trong hai ô đó rồi nhấn ENTER thì kích thước ma trận mới được cập nhật lại.

Kích hoạt <hộp mt> (xem hình), dùng các phím mũi tên hoặc chuột để di chuyển tới các vị trí trong ma trận. Dùng các phím ENTER hoặc F2 để sửa nội dung ô.

Để bổ sung ma trận vào danh sách, nhấn vào nút [<<] ở trên cùng của hàng nút nằm giữa hộp danh sách và hộp mt (xem hình). Chương trình tự động đặt tên cho ma trận. Nếu không hài lòng với cái tên này thì hãy nhấn 2 lần vào nó trong hộp danh sách để sửa lại tên (nhớ là nhấn chuột 2 lần chứ không phải là nhấp đúp).

Trên hàng nút này cũng có nút lưu một ma trận (biểu tượng đĩa mềm) và mở một ma trận (biểu tượng màu vàng).

### ***Các phép biến đổi cơ bản trên ma trận***

Được thực hiện ở <vùng biến đổi dòng, cột> và <vùng hoán vị dòng, cột> (xem hình). Hai khu vực này giúp bạn thực hiện tất cả các phép biến đổi cơ bản trên ma trận, (và cả những phép biến đổi quái chiêu khác nữa).

VD1:

Muốn đổi chỗ dòng 1 và dòng 3 của ma trận hiện tại (trong hộp ma trận) cho nhau thì nhập vào vùng hoán vị dòng cột theo khuôn dạng sau:

[D1] [<->] [D3]

Nháy vào nút [<->] để thực hiện phép đổi chỗ.

Lưu ý: các dấu ngoặc vuông trong các ví dụ là để hình tượng hoá nút bấm và các ô văn bản, đừng có nhập những dấu ngoặc vuông này vào!

Tương tự, muốn đổi chỗ cột 1 với cột 2 thì viết là

[C1] [<->] [C2]

Nháy vào nút <-> để thực hiện phép đổi chỗ.

Bạn có thể đổi chỗ một dòng với một cột cho nhau, nhưng kết quả thường rất khó nói, đặc biệt là khi ma trận không vuông.

VD2:

Muốn tính lại các phần tử trên dòng 1 bằng cách lấy dòng 2 nhân đôi rồi cộng với dòng 3. Nhập vào vùng biến đổi dòng, cột (xem hình) theo khuôn dạng sau:

[D1] [=] [D2] \* [2] + [D3] \* [ ]

hoặc

[D1] [=] [D2] \* [2] + [ ] \* [D3]

(bạn cứ viết thoải mái, không cần quan tâm đến thứ tự, ô nào không cần thì cứ bỏ trống ra)

Sau đó kích chuột vào nút [=] để thực hiện phép biến đổi

Tương tự nếu muốn dòng 1 bằng chính giá trị hiện tại của dòng 1 nhân với 2 nữa rồi cộng vào các giá trị cột 3.

[D1] [=] [D1] \* [2] + [C3] \* [ ]

Sau đó kích chuột vào nút [=] để thực hiện phép biến đổi

Nói chung bạn có thể nhập vào các ô văn bản trong <vùng biến đổi dòng, cột> một giá trị số thực hoặc một dòng/cột của ma trận (kiểu như D1, C3 ...), hoặc cũng có thể bỏ trống nó ra không nhập gì cả, nếu phải nhân với một ô trống, CT sẽ hiểu là nhân với 1, nếu phải cộng với một ô trống, chương trình sẽ hiểu là cộng với 0.

### ***Biến đổi nhanh ma trận***

Được thực hiện ở vùng biến đổi nhanh, tôi mới chỉ tạo ra hai công cụ:

Công cụ thứ 1: Dùng các phép biến đổi cơ bản để biến nửa dưới của ma trận về 0.

Công cụ thứ 2: Dùng các phép biến đổi cơ bản để biến 2 nửa của ma trận về 0.

Đây là 2 nút công cụ rất lợi hại giúp bạn tìm hạng ma trận, giải hệ phương trình bậc nhất bất kì (tối đa 255 ẩn, 255 phương trình).

Các phép biến đổi cơ bản trên ma trận mà 2 công cụ này sử dụng không có các phép biến đổi cột vì vậy bạn có thể an tâm mà dùng, tuy nhiên, nhiều khi kết quả sẽ hơi lẻ, nhưng độ chính xác khá cao đấy. Đặc biệt là,

nếu như kết quả ra lẻ mà lại có chu kỳ (thường nhìn thấy ngay nếu số liệu chẵn) thì bạn có thể xài công cụ hình kính lúp (đã giới thiệu ở phần trước) để đổi nó ra phân số.

### **Tính toán các ma trận**

Giả sử trong <hộp danh sách> đang có các ma trận A1, A2

VD1:

Gõ vào các <Ô vb1>, <Ô vb2>, <Ô vb3> (xem hình) nội dung lần lượt là "C", "A1", "A2"

Nếu cần tính ma trận  $C = A1 + A2$  thì kích chuột vào nút [+]

Nếu cần tính ma trận  $C = A1 - A2$  thì kích chuột vào nút [-]

Nếu cần tính ma trận  $C = A1 \times A2$  thì kích chuột vào nút [x]

Tương tự:

Nếu cần tính  $C = A1$  lũy thừa 3 thì:

- Nhập vào lần lượt là "C", "A1", "3",

- Kích chuột vào nút [^]

VD2:

Gõ vào các <Ô vbA>, <Ô vbB> (xem hình) nội dung lần lượt là "C", "A1"

Nếu muốn tính ma trận C là nghịch đảo của A1 thì kích chuột vào nút [^-1]

Nếu muốn tính ma trận C là chuyển vị của A1 thì kích chuột vào nút [^t]

### **Lưu ý**

- Khi tính toán ma trận, bạn có thể không cần nhập tên của ma trận kết quả (tức là để trống <Ô vb1>, <Ô vbA>), chương trình sẽ tự động đặt tên cho ma trận đích vì không thể quản lý ma trận mà không có tên (tất nhiên là bạn có thể thoải mái sửa lại tên ma trận). Tuy nhiên những cái tên chương trình tự động đặt thường khó nhớ. Vì vậy, tôi khuyến khích bạn tự đặt tên cho ma trận. Ví dụ bạn tính tổng ma trận "A1" và "A2" thì nên đặt tên cho ma trận kết quả là "A1+A2".

- Khi tìm chuyển vị hoặc nghịch đảo ma trận, bạn có thể bỏ trống luôn <Ô vbB>, khi đó chương trình sẽ tìm chuyển vị hoặc nghịch đảo của ma trận hiện tại (trong <Hộp mt>)

- Các ô của ma trận mang giá trị 0 hoặc có giá trị xấp xỉ 0 sẽ có màu chữ xám nhạt.

Tại sao lại phải đổi màu chữ những ô này?

Các số xấp xỉ bằng 0 thường rất kinh dị, ví dụ như 1.73472347597681E-18, (thực sự thì số này rất rất xấp xỉ số 0, đọc lại phần số viết theo dạng khoa học). Nếu bạn nhìn thấy cả con số dài ngoằng trên và thấy cái đuôi E-18 của nó thì bạn sẽ biết ngay nó xấp xỉ 0, không có vấn đề gì. Nhưng mỗi ô con trong <Hộp mt> không đủ dài để nhìn thấy toàn bộ con số trên nên cái mà bạn thấy trong ô đó chỉ là mấy số ở phần đầu mà thôi, đó là 1.734723. Con số đó rất dễ gây hiểu lầm là ô đó mang giá trị 1.7 chứ không phải giá trị 0, trong khi giá trị thực sự của nó xấp xỉ 0 (gần như là 0 luôn). *Vì vậy khi bạn thấy một ô có chữ màu xám (nhạt hơn các ô khác), dù trông lè đến đâu nhưng bạn có thể coi ô đó mang giá trị 0 luôn, không cần phải quan tâm.*

- Nếu muốn nhìn thấy rõ toàn bộ nội dung ô, bạn phải di chuyển đến đó, nhấn F2 hoặc ENTER để vào chế độ sửa ô rồi mới xem được toàn bộ nội dung ô (hoặc cắt dán nội dung ô ra chỗ khác rộng hơn mà xem).

Điều này khá phiền phức. Tôi đã khắc phục như sau:

Trước hết, di chuyển con trỏ chuột ra ngoài <Hộp mt> (đâu cũng được, miễn là ra ngoài). Bây giờ lại di chuyển nó trở vào ô cần xem nội dung trong <Hộp mt>, đợi một lúc sẽ thấy dưới con trỏ chuột có một hình chữ nhật bên trong có 2 dòng: Dòng đầu là tên ma trận hiện tại, dòng 2 là nội dung ô nằm dưới con trỏ. Khi muốn biết ma trận hiện tại trong <Hộp mt> mang tên gì cũng dùng cách này.

- Các ô rỗng mang giá trị mặc định là 0 (Nếu như bạn mở rộng số dòng, cột của ma trận thì sẽ hình thành các ô rỗng, nếu bạn không nhập giá trị vào đó thì chương trình sẽ coi nó là giá trị 0).

## Bài tập

Để chắc chắn bạn đã làm chủ được sức mạnh của chương trình hãy làm thử giải quyết nhanh gọn các bài tập sau:

**1)** Trong các số sau, số nào là số nguyên tố? Nếu không phải là số nguyên tố thì hãy phân tích nó ra thừa số nguyên tố: 12345; 2003; 1984; 95256.

**2)** Tìm ước chung lớn nhất và bội chung nhỏ nhất của:  $x*4$  và  $x+3$ ; với các giá trị  $x$  lần lượt là:

a, 3

b, 9

c, 1997

d, 2003

**3)** Còn bao nhiêu ngày nữa là sinh nhật 80 tuổi của bạn?!

Hôm đó là thứ mấy?

Bao nhiêu ngày trôi qua kể từ ngày 1/1/0000 cho tới ngày hôm đó?

Năm đó có phải là năm nhuận không?

Năm đó là năm con gì, can chi của tháng đó là gì?

**4)** Tìm giá trị chân lý của:

$T \leftrightarrow (x \leq 3) \rightarrow \neg ((x \leq 5) \vee (x \geq 2) \rightarrow T)$

với  $x$  là:

a,  $\pi^2 - 1$

b,  $-\pi$

**5)** Đổi sang hệ cơ số 7 và cơ số 12 các số sau:

a,  $\overline{432101234}_5$

b,  $\overline{ABC0ABC}_{14}$

**6)** Tìm dạng phân số của các số hữu tỉ sau:

a, 0.19 84 84 84...

b, 0.8 123 123 123...

c, 0. 2003 2003 2003...

**7)** Tìm các nghiệm gần đúng của phương trình:

$$a, x \sin^2 x + \cos x = 0 \quad x \in (-\pi.. \pi)$$

$$b, x + \operatorname{tg}(x^2) = 0 \quad x \in (-1..1)$$

$$c, \ln(x^2) - x + 1 = 0 \quad x \in \mathbb{R}$$

**8)** Cho ma trận:

$$A = \begin{bmatrix} 4 & 5 & 6 & 7 \\ 3 & -4 & -5 & 8 \\ 2 & -3 & -6 & 9 \\ 1 & -2 & -1 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 1 & 2 & -2 & -4 & 5 & 6 \\ 3 & 1 & 4 & -5 & 0 & 0 \\ 4 & 1 & 2 & 9 & 0 & 0 \\ -1 & -2 & 0 & 2 & 3 & 4 \\ -2 & 4 & 0 & 2 & -3 & 0 \\ 0 & -2 & 0 & 4 & 2 & 2 \end{bmatrix}$$

Tính định thức và tìm chính xác ma trận nghịch đảo của các ma trận trên (không chấp nhận giá trị xấp xỉ)?

Tính  $\det A^2$  và tìm ma trận  $B^3$

**9)** Giải hệ:

$$\begin{cases} 3x_1 - 2x_2 + 3x_3 - x_4 + 4x_5 - 3x_6 = -2 \\ 2x_1 + x_2 - 2x_4 - 2x_5 + 4x_6 = 11 \\ x_1 - 2x_2 + 2x_3 - 3x_4 + 4x_5 - 5x_6 = -7 \\ x_2 + 2x_3 - 4x_5 - 5x_6 = -21 \\ 2x_1 - 2x_3 + 3x_4 - 4x_5 + 2x_6 = -13 \\ 4x_1 - 2x_2 + x_3 - 2x_5 + x_6 = -7 \end{cases}$$

Gợi ý: Dùng phương pháp Gauss giải ra:  $x_1=-2$ ;  $x_2=1$ ;  $x_3=3$ ;  $x_4=-1$ ;  $x_5=2$ ;  $x_6=4$

Trên đây chỉ là vài bài tập cũ xì. Nếu bạn có bài tập nào nghĩ ra cách ứng dụng chương trình một cách độc đáo xin hãy gửi về cho tôi.

## Hướng tới tương lai

Đây là những tiện ích tôi chưa làm kịp:

- Bổ sung thêm kiểu số phức, các hàm và phép toán liên quan (ta đã có các kiểu: số, phân số, chuỗi, ngày/giờ, logic)

- Viết mục giúp đỡ (Help) cho tử tế và chi tiết (chứ không dồn hết vào một tài liệu như thế này).

- Tìm đạo hàm của hàm số.
- Đổi các đơn vị đo lường.
- Lập bảng giá trị chân lý cho các biểu thức Logic.
- Tính toán với các số nguyên hàng trăm chữ số.
- Hoàn thiện các thao tác với ma trận, hoàn chỉnh giao diện chung, nâng cấp tốc độ vẽ đồ thị, thêm history list...
- Bổ sung việc tính toán với lịch Julia (dùng trong thiên văn học)
- Các hàm tính sai số và thao tác với dãy số.
- Quy đổi lịch âm dương. Thực sự tôi chưa tìm thấy thuật toán nào có thể đổi qua lại giữa 2 hệ thống lịch này. Nếu bạn có xin hãy chỉ cho tôi biết, Thanks

Chương trình chắc chắn còn sai sót. Nếu bạn phát hiện được, xin liên hệ theo địa chỉ dưới để kịp thời sửa chữa. Cảm ơn nhiều!

Dương Phú Hiệp  
Lớp T15B2 - trường ĐHDL Thăng Long - Hà nội  
ĐC: 77 Nguyễn Hữu Huân Q.Hoàn Kiếm Hà Nội  
ĐT: 9260551  
Email: duongphuhiep@hotmail.com