Enjoy reading the January issue of the HP-39/40g newsletter!

In this issue we will talk about:

- 1. Letter from the editor.
- 2. Program of the month.
- 3. Upcoming Programs.
- 4. Interview with Nobody.
- 5. Michaël's cosy programmers corner (NEW!)

Letter from the editor.

First of all: a happy new year to all of you!

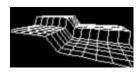
Here in Belgium at the start of a new year we have the custom to make good resolutions. Stop smoking, lose weight or something else. Mine is that I should start writing this newsletter much earlier than a day or two before the new month[©] and then the monthly newsletter would be on time for a change!

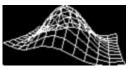
Another good intention would be that I introduce some new topics in this newsletter. I've noticed that most of the Upcoming programs take a long time to get finished or that is cancelled. I don't want the people to think that they read bullshit or that I invent these programs. So I've decided to make some new topics.

And last good intention is giving the newsletter a new lay-out. I will change the lay-out, gradually of this newsletter in the couple of months.

But for now enjoy reading the January edition of the newsletter.

Program of the month.







This month the 'program of the month' is a 3D graphing utility. Yet though one that is in HP-basic. So it's rather slow. 5-6 minutes for a 3D graph is not exactly what you are calling fast. But because it actually works and does what it is supposed to do without too much rubbish (I'm not very keen on seeing messages/logos and etc. that hasn't anything to do with the math program itself). Only drawback is that it's in French. Though it is very easy to make the connection to the English equivalent words. The program itself is very easy to operate. Put your desired function in F1(X) and run the program. You can either choose between AUTO or between MANUAL. When choosing for AUTO the program does it all by itself and with the other option you have to answer some questions. After that the 3D graph gets plotted and the result is stored in grob G0. As for a conclusion: for everybody who needs a 3D graphing program. This is the best one available for the moment for the HP-39(+)/40g.

Specs:

System: HP 39(+)/40g. Author: Hydreprod

Download where: http://perso.wanadoo.fr/ring.avalon/hydreprod/Real3D.zip

Site of the maker (French): http://www.hydreprod.fr.st

Upcoming programs.

A big Rumour with a capital R. Some people say that HP-Poska is trying to convert EQVLIB16 (or whatever it's called). I say don't count on it.

"Family None" on the comp.sys.48 group: "I am working on a diss/assembler for Aplets/E-leasons for the calculators". Yes, he was talking about the HP-38/39g. Let's hope he succeeds ©.

Mackenzie's current project is a HP-Basic to Sys-RPL converter. A very interesting project if you ask me! Let's hope he succeeds too! An other project from him is a 'Visual Basic' style of development environment for the 39G, where little or no coding is required to make apps and games.

U-Lib from NODA once listed first in most wanted programs (see July issue 2003) is cancelled. However he will release his commented sources someday... Whenever that may be (he's not exactly fast) I will translate them to English.

Because I didn't program in a while, I decided to make a black jack game to refresh my memory.

I'm also making an empty aplet (like JYA or NODA) but with the views enabled. WOW, now that I look back to all my projects that I still have to finish. I'm sure to have work till 2010 ©.

Interview with nobody.

No interview this month ③. Only some corrections to the last one with Lillian Pigallio. He had given some links to websites which I forgot to add. To be complete I will give them to you now:

www.essi.fr

www.supralog.com

www.amadeus.net

http://www.beyond.fr/villages/tourrettesloup.html

http://www.lexpress.fr/Express/Info/Societe/Dossier/village/dossier.asp?ida=342807

http://mo6.free.fr/author3.html (emulator)

However I think that most of these links are in French so probably 80% of the readers of this newsletter will not understand (much) what's on these sites.

Next month again an interview.

Michaël's cosy programmers' corner.

What's this? A new topic? Yes it is! Like I told before I noticed that most of the upcoming programs get cancelled or take a long time to get finished. I don't blame people for that. Programming calculators is not the only thing in the world. But with this topic I want people to urge to program their calculators. I will explain how do certain things in HP-Basic and in Sys-RPL. E.g.: this month I will explain how to do a fast getkey in Sys-RPL. I recommend though that you have read my sys-rpl tutorials (which BTW are recently updated). The link is: http://www.hpcalc.org/hp39/docs/sysrpllessons.zip. Things will be little bit less detailed explained as in my Sys-RPL lessons. But someone with a little bit perseverance should be OK!

Anyway Let's start:

Make an aplet with the name 'Getkey' and the same file name in WSR 1.1b. In the editing window, scroll down till the beginning of the program and delete the usual lines. First make two variables called XPix and YPix. Store BINT10 in each of them. Clear the screen and set a pixel with XPix and YPix as coordinates. Now we are going to make a loop. Why? You might ask. A good question! Because GETTOUCH unlike WaitForkey doesn't wait till a key is pressed. WaitForkey? Yes, that is a slower getkey in Sys-RPL, but till now we only used it only for a 'pause' in our program. So let's make a loop: first write BEGIN. And under that write UNTIL. Under BEGIN we are going to say to the compiler when there is no key pressed that he has to put ZERO on the stack. Why we have to do that? Because every key has a certain number. Look in Head.39 for those keys and their numbers.

E.g.: DEFINE kcSoftKey3 THREE. If there is already number on the stack and it the same as one of the number of the key you want to check for then you get a weird situation. So under BEGIN write GETTOUCH NOT IT ZERO. Your code should like this:

```
HARDBUFF ZEROZERO BINT_131d SIXTYFOUR GROB!ZERODRP

BINT10 DUP XPix! YPix!
XPix@ YPix@ PIXON

BEGIN

GETTOUCH NOT IT ZERO

UNTIL
```

Let's say we want to quite the program when the key DEL is pushed. In head.39 we see that DEL is equal to kcDel. Above until we write kcDel Repkey? The Repkey? says that when the key is pushed it has to repeat the code. Nothing prevents you from using BINT21 instead of kcDel (because kcDel is defined in head.39 as BINT21) Now let's move the Pixel with the Right Arrow. the code for the Right Arrow is indeed kcRightArrow. So when kcRightArrow is pushed we have to set the pixel off, update the coordinates, and turn the pixel back on. Write under Gettouch not it zero kcrightArrow repkey? Beneath that do an IF THEN test. Your code should be like this:

```
BEGIN

GETTOUCH NOT IT ZERO

kcRightArrow REPKEY?

IT
::
;

kcDel REPKEY?

UNTIL
```

Now turn the pixel off, add one to the variable XPix and set the pixel back on. Here is the final code:

```
BEGIN

GETTOUCH NOT IT ZERO

kcRightArrow REPKEY?

IT
::
XPix@ YPix@ PIXOFF
XPix@ #1+ XPix!
XPix@ YPix@ PIXON
;

kcDel REPKEY?
UNTIL
Entry
```

Now compile the code and see that pixel move! If anyone wants to see an other way for a fast getkey (code written by Andri Winati) can always mail me at the normal address (ndco232@tiscalinet.be). Also if have you any suggestions, ideas for the next issue of the monthly newsletter than you may mail me too.