

Hail

V 1.05

Por Samuel Stearley

Implementación en Programas TI-Basic Manual hecho por Raul Bores Monjiote (izerw)

El primer paso para implementar el Hail en una programación, es verificar que efectivamente se encuentre en la calculadora antes de ejecutarlo, y de caso contrario advertir al usuario que el archivo no se encuentra.

El Hail consta de 2 archivos:

- Hail
- Haildata

A continuación se muestra el código para validar el Hail, éste código debe ir al principio del programa:

```
: Prgm
:   If getType(hail)="NONE" or getType(haildata)="NONE" Then
:       Dialog
:           Title "ERROR"
:           Text "Falta el Hail o Haildata"
:       EndDlog
:       DispHome
:       Return
:   EndIf
:
:   <<Programa>>
:
: EndPrgm
```

De hecho si estuviéramos usando la librería FLIB, podríamos reemplazar la ventana de Dialogo por lo siguiente:

```
:       flib("msg:ERROR|Falta el Hail o Haildata")
```

Ahora para ejecutar el Hail y poder extraer la expresión escrita usamos el siguiente código:

```
: Local x
: ClrIO
: hail()
: Disp "           Presione ENTER"
: Disp "           para continuar"
: InputStr "",x
: expr(x) → x
```

Lo que hace el Hail es “transferir” la expresión escrita al siguiente campo de texto.

También podemos emplear una ventana de Dialogo para introducir alguna expresión:

```
: Local x
: hail()
: Dialog
:   Title "Presione ENTER para continuar"
:   Request "",x
: EndDlog
: expr(x) → x
```

Como recomendación, utilice un sistema para validar datos, en éste caso validar “x” para comprobar si es una expresión o un dato valido para el programa.

Argumentos

Ésta opción, permite poder introducir alguna expresión al Hail.

Con un argumento:

Si solo disponemos de un argumento, se abrirá el Hail, en él se mostrará la expresión y podemos modificarlo posteriormente, hay dos formas de introducir la expresión:

`hail(expr)` El argumento es simplificado por la calculadora antes de que se pase al hail como argumento.

`hail("expr")` El argumento es una cadena (string) así que no es simplificado por la calculadora, y pasa directamente al Hail.

Con 2 argumentos:

`hail(expr, fuente)` La expresión puede ser una cadena o una expresión normal. Si se pasa una cadena el argumento no será simplificado. La fuente es el tamaño de letra utilizar, 0 para letra pequeña y 1 para letra normal.

En éste modo, el Hail no se abrirá, ni mostrará ninguna expresión en la pantalla, lo único que hace es guardar en variables el ancho y alto de la expresión (en PrettyPrint), en las siguientes variables:

xtop	La mitad superior
xbottom	La mitad inferior
xwidth	La anchura

Estas variables no se deben archivar, por lo tanto, si se ejecuta el Hail en éste modo ponga estas variables como variable **Locales**.
Para tener el alto total de la expresión solo se debe sumar xtop y xbottom.
Las medidas son dadas en Pixeles.

Con 3 argumentos:

`hail(expr, x, y)` A partir de 3 argumentos, el Hail ya no se abrirá, por lo que la expresión ingresada no podremos modificarla posteriormente, en vez de eso, el Hail imprimirá la expresión directamente en la pantalla de PrgmIO en las coordenadas x, y. Las coordenadas pueden ser negativas; expr puede ser una cadena o una expresión normal.

Si la expresión es demasiado grande y sobrepasa los límites de la pantalla, la expresión se mostrará correctamente, pero será recortada; para ver la totalidad de la expresión, vea el siguiente argumento o use el eviwer (su funcionamiento se explica mas adelante)
El tamaño de letra que se usa es la normal.

También guardará las dimensiones de la expresión en las variables dadas arriba.

Ejemplo:

```
:      Local  xtop,xbottom,xwidth
:      ClrIO
:      Disp
:      hail("2y/3",10,10)
:      Pause
```


Con 4 argumentos:

`hail(expr, x, y, fuente)` Se emplea de la misma forma que el anterior (3 argumentos) con la única diferencia que se puede escoger el tamaño de letra que se va a imprimir.
El tamaño de letra a utilizar es, 0 para letra pequeña y 1 para letra normal.

Este último argumento es bastante útil, ya que podemos imprimir expresiones muy grandes en letra pequeña para que alcance en toda la pantalla.

Extras

hailCom

El hailCom es un programa que viene junto con el Hail, y tiene la utilidad de poder guardar alguna expresión en el portapapeles (clipboard) de la calculadora, el cual podemos recuperar con +ESC.
La expresión debe ser una cadena.

Ejemplo:

```
: 2y+3 → x
: hailcom(string(x))
```

De ésta forma la calculadora evaluará primero la expresión antes de pasar a hailCom.

```
: hailcom("2y+3")
```

De ésta forma la expresión pasará directamente a hailCom.

eviewer

El eviewer es un programa hecho en TI-Basic el cual usa el Hail en el modo de 4 argumentos, con el objetivo de poder imprimir cualquier expresión, pero permite mover ésta expresión por toda la pantalla, por ejemplo si tenemos una expresión demasiado grande, podemos visualizar su totalidad. El tamaño de letra por default es 0.

Se ejecuta de la siguiente forma:

```
eviewer(expr)
```

Pudiendo expr ser una expresión o una cadena. Si es una expresión será simplificado por la calculadora antes de imprimir, y si es cadena lo mostrará sin simplificar.

Para poder movernos se usa las siguientes teclas:

8	Arriba
2	Abajo
4	Derecha
6	Izquierda

Para salir presione ESC.

Cualquier duda o comentario:

Escriba a: izerw1@gmail.com

ó visite el foro de calculadoras.cl dedicado a las TI:
<http://www.calculadoras.cl/foro/forumdisplay.php?f=2>