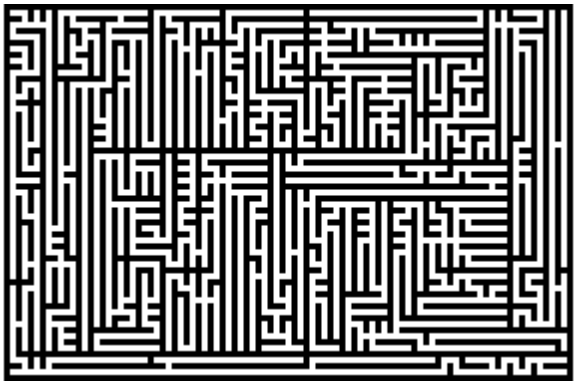
Daniel De Graaf's MazePack 2.5

This MazePack is made for MirageOS, although it can be run from TI's home screen. A description of each program and its function follows:

- Make

  Makes a random maze and stores it in Pic1. Displays the maze as it is being created. You can create your own mazes: draw lines on the graph, and store them to Pic1. Then, store $3+5i$ to A and run Make to finish the maze. Make turns off the calculator after it is finished to preserve battery life. An example of a finished maze:



- Solve

  Solves the maze in Pic1 using the dead-end filler method. This leaves a single path with black space around it. When it is finished, Solve stores the solution in Pic2, turns off the calculator, and pauses the maze.

  Note: If you created your own maze, it may have multiple solutions. Solve shows all solutions.

- View

  If the maze and solution exist, they are grayscaled together on the screen – the maze is grey and the solution is white. Then, it asks if you want to delete both, the solution, or none

  If no solution exists, View displays the maze on your screen, and then asks if you want to delete it.

- Play

  Plays the maze in Pic1. You start in the upper-left corner and finish in the lower right. You and the part of the maze directly around you are displayed on the screen. Keys:

  Arrows – move 1 space, or 2 pixels.

  8, 4, 6, and 2 – move to the end of a passage.

  5 or APPS – display a map of the maze with a blinking dot at your location.

  DEL – restart maze

  Clear – Turn calculator off.

  2^{nd}+Quit – Save and exit.

  Note: Your position is encoded in the list MAZEP. This list is encoded, and if changed will restart the maze.

The MazePack also needs 7 assembly programs in RAM – They start with θ so they are at the end of your program list.
θAPDØ – Disables automatic shutoff
θAPD1 – Enables automatic shutoff
θPower – Turns off calculator
θRunØ – Turns off run indicator
θGetKey – Enhanced GetKey routine. ON Key disabled, waits for keypress, automatically stores key in θ, allows 2nd and Alpha keys.
θGrey – Greyscales the maze and solution (Pic1 and Pic2) – you can use
θDisp – Displays any character on screen. To use:

1. Store *row**256+*column* to X
2. Store *character* (from 0 to 255) to Y
3. Run Asm(prgmθDisp)

Make sure that *row* is from 1 to 16, *column* is from 1 to 8, and *character* is from 0 to 255. If they are not, the calculator will crash. If it crashes:

1. Take out 1 battery
2. Hold down Clear key
3. Replace battery
4. Hold down ON key
5. Release ON and Clear keys

This should bring you to the home screen. If you're unlucky, it will also clear your RAM (although this has not happened to me).

To use the θ routines in your program type Asm(prgm*name*).

If you use any of the θ routines, please give me credit.