



1. Edit display is indicated by highlighted tab.
- 2, 3. Typical edit and execute operations are available through icons and menus, dimmed-out when unavailable or not applicable.
4. WYSIWYG editor for MIPS assembly language code.

The screenshot displays the MASM68000 IDE interface. The main window shows assembly code with several lines highlighted in yellow. Annotations 1 through 9 point to specific features:

- 1:** Points to the **Execute** button in the top toolbar.
- 2:** Points to the **Labels** tab in the **Registers** window.
- 3:** Points to the **Value (+4)** column in the **Data Segment** window.
- 4:** Points to the **Hexadecimal Values** checkbox in the **Data Segment** window.
- 5:** Points to the **Run speed at max (no interaction)** slider in the **Registers** window.
- 6:** Points to the **Labels** tab in the **Registers** window.
- 7:** Points to the **Value (+18)** column in the **Data Segment** window.
- 8:** Points to the **Registers** window.
- 9:** Points to the **Labels** tab in the **Registers** window.

1. Execute display is indicated by highlighted tab.
2. Assembly code is displayed with its address, machine code, assembly code, and the corresponding line from the source code file. (Source code and assembly code will differ when pseudoinstructions have been used.)
3. The values stored in Memory are directly editable (similar to a spreadsheet).
4. The window onto the Memory display is controlled in several ways: previous/next arrows and a menu of common locations (e.g., top of stack).
5. The numeric base used for the display of data values and addresses (memory and registers) is selectable between decimal and hexadecimal.
6. Addresses of labels and data declarations are available. Typically, these are used only when single-stepping to verify that an address is as expected.
7. The values stored in Registers are directly editable (similar to a spreadsheet).
8. Breakpoints are set by a checkbox for each assembly instruction. These checkboxes are always displayed and available.
9. Selectable speed of execution allows the user to "watch the action" instead of the assembly program finishing directly.