

```

.globl store_var # Make the `store_var` label globally accessible.
.globl store_loop # Make the `store_loop` label globally accessible.
.globl store_here # Make the `store_here` label globally accessible.
# Routine to store a variable in the symbol table.
store_var:
la t0, symbol_table # Load the address of the symbol table into t0.
store_loop:
lw t1, 0(t0) # Load the current slot's value (first word in the slot).
beqz t1, store_here # If the slot is empty (contains 0), jump to
`store_here` to store the variable.
addi t0, t0, 8 # Move to the next slot (assuming each slot is 8 bytes:
4 for the name, 4 for the address).
j store_loop # Repeat the loop to check the next slot.
store_here:
lb t1, 0(a0) # Load the first character of the variable name from the
input in `a0`.
sw t1, 0(t0) # Store the variable name into the current slot's first
word.
sw a1, 4(t0) # Store the address (passed in `a1`) into the second word
of the current slot.
ret # Return to the caller.

```