The hatchet shell is a basic command line interpreter. This shell has no levels or file systems and presents a basic ">" character only.

The shell stores each entered line as a string of char that parses on the return key depression. Other keys that cause edit are as follows:

up:	bring up previously entered line (unfinished line is saved but not processed)
down:	bring up subsequent line
left:	Move cursor left, stopping at left-most char plus 1
right:	Move cursor right, stopping at line length
Typing:	Overwrite mode only
Backspace:	change previous char to space
return:	Parse line

Up or down at top or bottom rolls around

Command buffer:
cmdline[8][40]Consists of 8 memory lines, 40 char deep, for a total of 120 bytes. [0][0] through [0][39] is a continuous block of memory, as are each subsequent string of 40.
cmd_entered_ptr: is type char * and emulates a string at the last entered line, at parse time.

Comparison and equation of commands/strings
CASE A
When we press return expecting say, "larson", we have assumptions of what is qualified on an ascii stream level. The following notation is that anything in brackets is 8 bits.

- [l][a][r][s][o][n][any alpha char]: Rejected
- [l][a][r][s][o][n][0x00]: Accepted
- [l][a][r][s][o][n][space]: Accpeted
- [l][a][r][s][o][n][0x0A]: Should not happen
- [l][a][r][s][o][n][0x0D]: Should not happen
- "larson" Is also case sensitive

This is one interpretation of sameness.

CASE B
An alternate use is to compare an arbitrary block of any string. Consider: "abcdefg12345678". To determine if the second eight characters are "12345678", the function needs to have a comparison string passed as well. In this case, it doesn't matter what surrounds the text. This could incorperate wildcards but it doesn't.

To simplify the matter, only pass two items, two char * variables into type name[] places. The function evaluates and returns:

- 0 for both case A and B have no sameness
- 'Not 0' for case A or B registers sameness

The function is determined by the start and length arguments.
If both zero, the null termination is used to determine length. This method requires either space or null term to match.
If the length is an integer, the must match regardless of ascii interpretation of the data/termination.
If length zero the output is always false.