Questions?

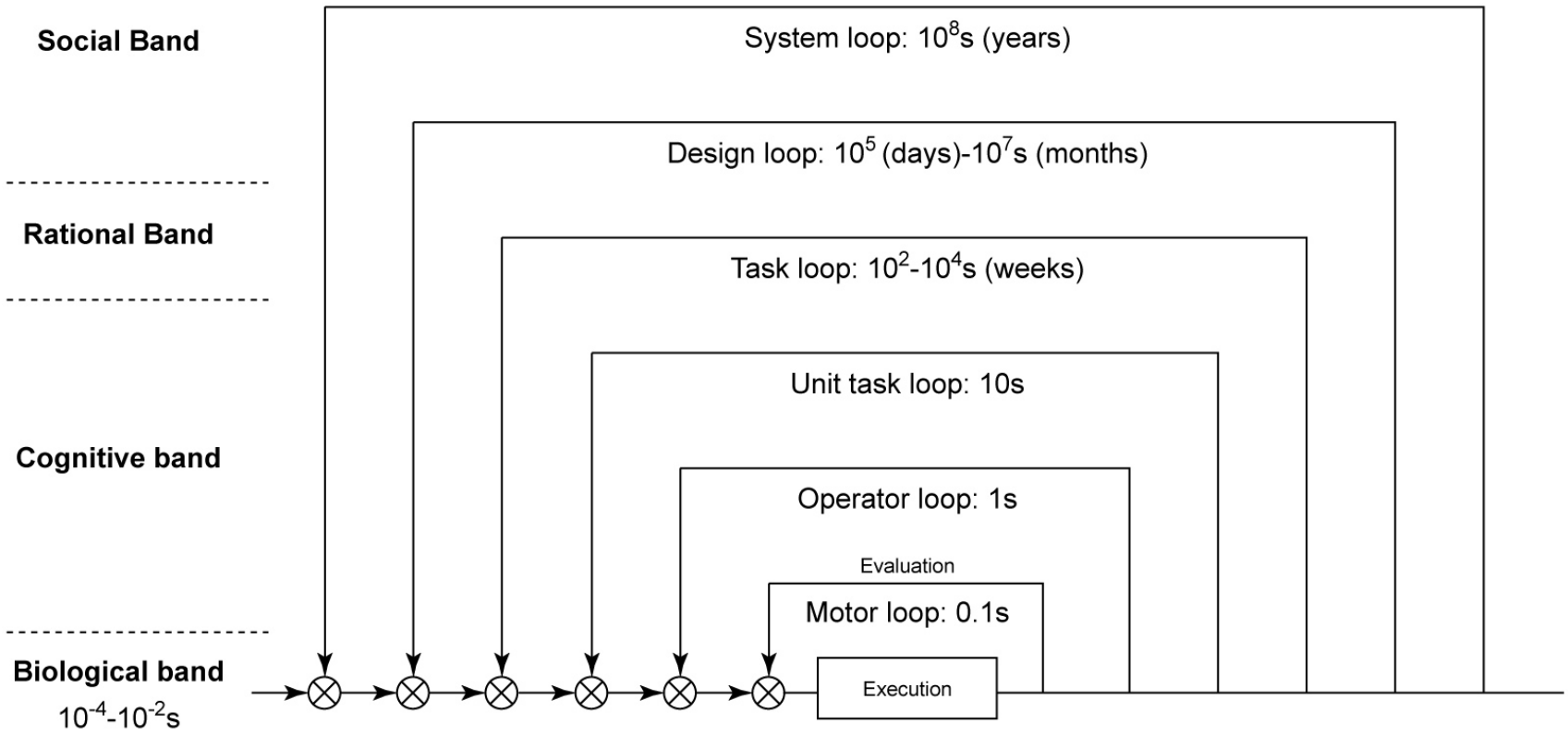
- Project #2
- HW #3 in
- HW #4 out
- Flash tutorial this Friday

Survey

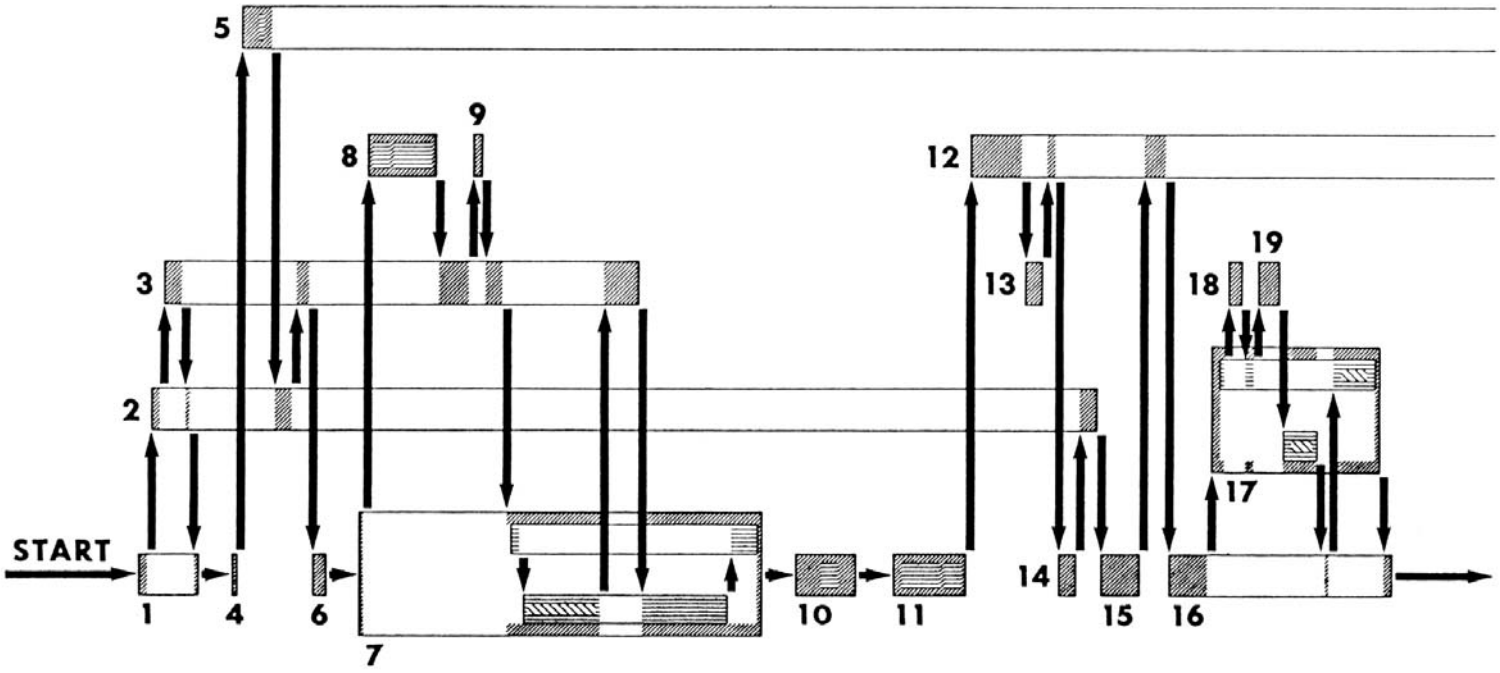
- Printing
- Workload
  - Reading
  - Homework
    - *Vague*
    - *Timely feedback*
- Class participation

- Comments
  - All slides are online
  - Use office hours if you have questions
  - New study session (see web page)
  - Midterm and final

Human interaction loops (Newell)



Task structure

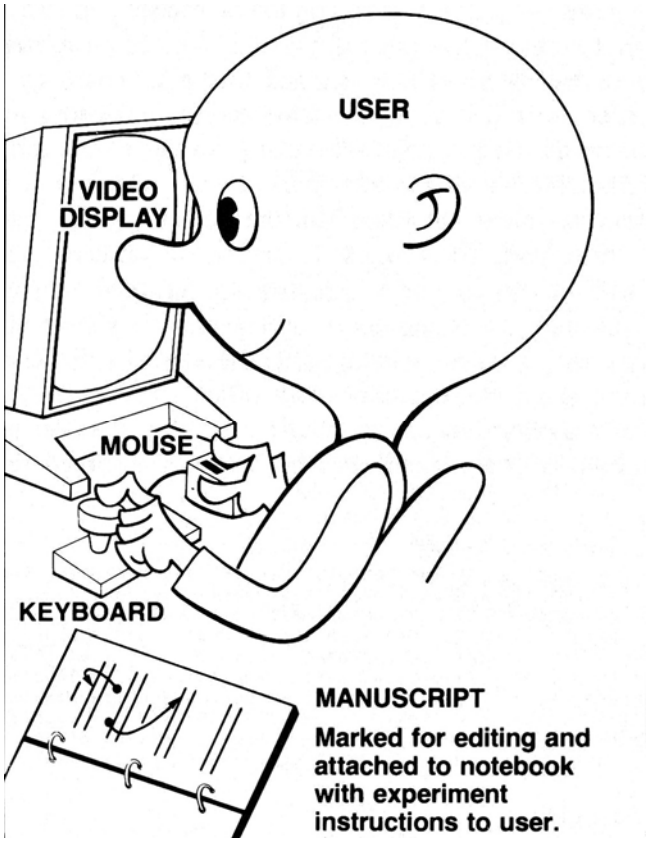


GOMS (Card et al.)

- Family of methods
  - KLM, CMN-GOMS, NGOMSL, CPM-GOMS
- Describe the user behavior in term of
  - Goals
    - *Edit manuscript, locate line*
  - Operators
    - *Elementary perceptual, motor or cognitive acts*
  - Methods
    - *Procedure for accomplishing goals*
  - Selection rules
    - *Used if several methods are available for a given goal*

GOMS example I

- Setting



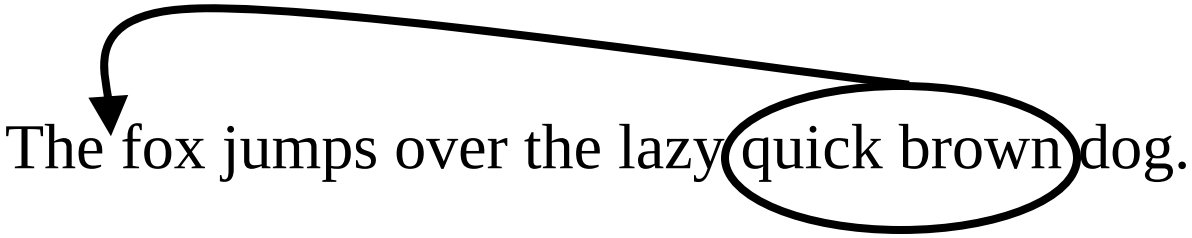
- Analysis

GOAL: EDIT-MANUSCRIPT
. GOAL: EDIT-UNIT-TASK *repeat until no more unit tasks*
. . GOAL: ACQUIRE-UNIT-TASK
. . . GET-NEXT-PAGE *if at end of manuscript page*
. . . GET-NEXT-TASK
. . GOAL: EXECUTE-UNIT-TASK
. . . GOAL: LOCATE-LINE
. . . . [select: USE-QS-METHOD
USE-LF-METHOD]
. . . GOAL: MODIFY-TEXT
. . . . [select: USE-S-COMMAND
USE-M-COMMAND]
. . . . VERIFY-EDIT .

GOMS example II

(From HCI Models, Theories and Frameworks, J. Carroll)

- Using a text editor edit the following text as shown

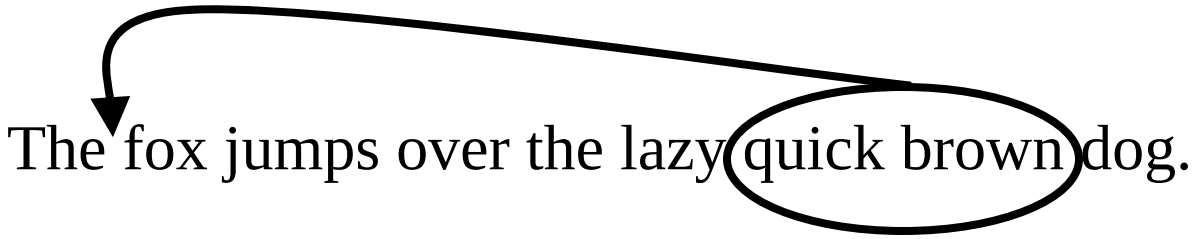


- Goals and sub-goals?
- Operators?
- Methods?
- Selection rules?

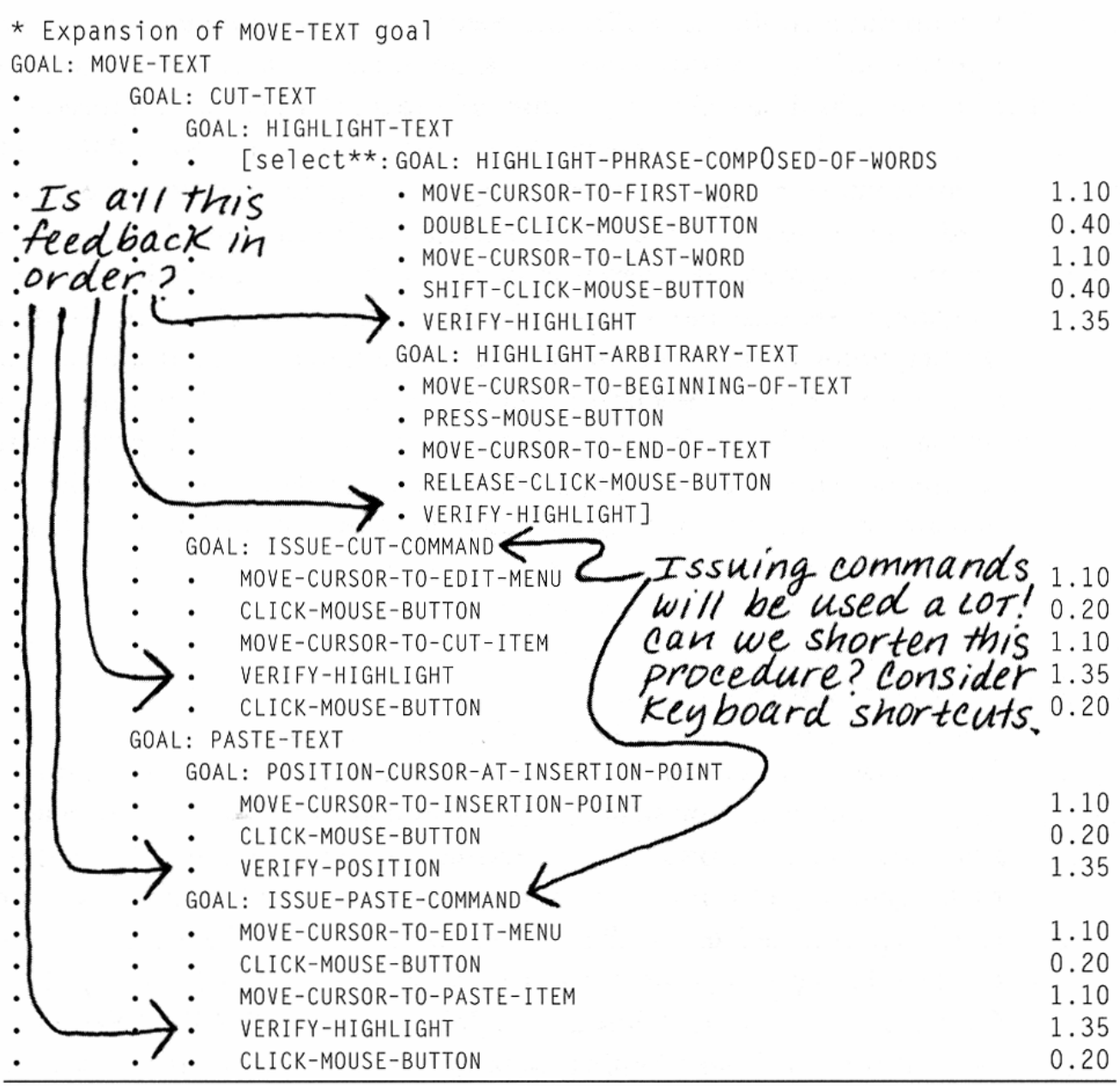
GOMS example II

(From HCI Models, Theories and Frameworks, J. Carroll)

- Using a text editor edit the following text as shown



- Analysis?


* Expansion of MOVE-TEXT goal
GOAL: MOVE-TEXT
GOAL: CUT-TEXT
GOAL: HIGHLIGHT-TEXT
[select**: GOAL: HIGHLIGHT-PHRASE-COMPOSED-OF-WORDS
• MOVE-CURSOR-TO-FIRST-WORD 1.10
• DOUBLE-CLICK-MOUSE-BUTTON 0.40
• MOVE-CURSOR-TO-LAST-WORD 1.10
• SHIFT-CLICK-MOUSE-BUTTON 0.40
• VERIFY-HIGHLIGHT 1.35
GOAL: HIGHLIGHT-ARBITRARY-TEXT
• MOVE-CURSOR-TO-BEGINNING-OF-TEXT
• PRESS-MOUSE-BUTTON
• MOVE-CURSOR-TO-END-OF-TEXT
• RELEASE-CLICK-MOUSE-BUTTON
• VERIFY-HIGHLIGHT]
GOAL: ISSUE-CUT-COMMAND
• MOVE-CURSOR-TO-EDIT-MENU 1.10
• CLICK-MOUSE-BUTTON 0.20
• MOVE-CURSOR-TO-CUT-ITEM 1.10
• VERIFY-HIGHLIGHT 1.35
• CLICK-MOUSE-BUTTON 0.20
GOAL: PASTE-TEXT
GOAL: POSITION-CURSOR-AT-INSERTION-POINT
• MOVE-CURSOR-TO-INSERTION-POINT 1.10
• CLICK-MOUSE-BUTTON 0.20
• VERIFY-POSITION 1.35
GOAL: ISSUE-PASTE-COMMAND
• MOVE-CURSOR-TO-EDIT-MENU 1.10
• CLICK-MOUSE-BUTTON 0.20
• MOVE-CURSOR-TO-PASTE-ITEM 1.10
• VERIFY-HIGHLIGHT 1.35
• CLICK-MOUSE-BUTTON 0.20
TOTAL TIME PREDICTED (SEC) 16.25
Is all this feedback in order?
Issuing commands will be used a LOT! Can we shorten this procedure? Consider keyboard shortcuts.

Keystroke Level Model (KLM)

- Describe the task using the following operators:
  - K: pressing a key or a pressing (or releasing) a button

    t_K = *0.08 - 1.2s*
  - P: pointing

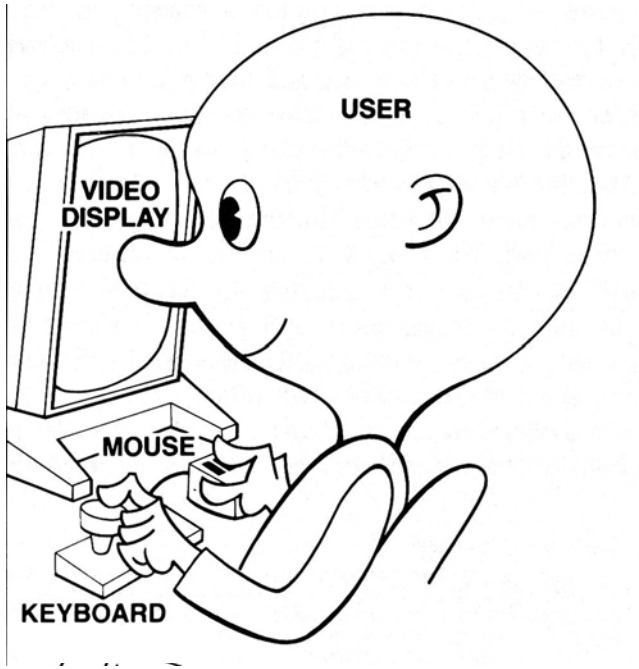
    t_P = *1.1s (without button press)*
  - H: Homing (switching device)

    t_H = *0.4s*
  - D(*n*,*l*): Drawing segmented lines

    t_D = *0.9*n + .16*l s*
  - M: Mentally prepare

    t_M = *1.35s*
  - R(*t*): system response time

    t_R = *t*



How to use KML

- Encode using all physical operator (K, P, H, D(n,l), R(t))
- Apply KML rules [0-4]
- Transform R followed by an M
  - If $t \leq t_M$: R(t) $\rightarrow$ R(0)
  - If $t_M < t$: R(t) $\rightarrow$ R($t - t_M$)
- Compute the total time by simply adding all times
  - Will describe expert user behavior

KLM heuristic rules (Raskin's)

0: Insert M

- *In front of all K*
- *In front of all P's selecting a command*

1: Remove M between *fully anticipated* operators

- *PMK → PK*

2: if a string of MKs belong to *cognitive unit* delete all M but first

- *4564.23: MKMKMKMKMKMKMK → MKKKKKKK*

3: if K is a *redundant terminator* then delete M in front of it

- *↵↵: MKMK → MKK*

4a: if K terminate a constant string (command name) delete the M in front of it

- *cd↵: MKKMK → MKKK*

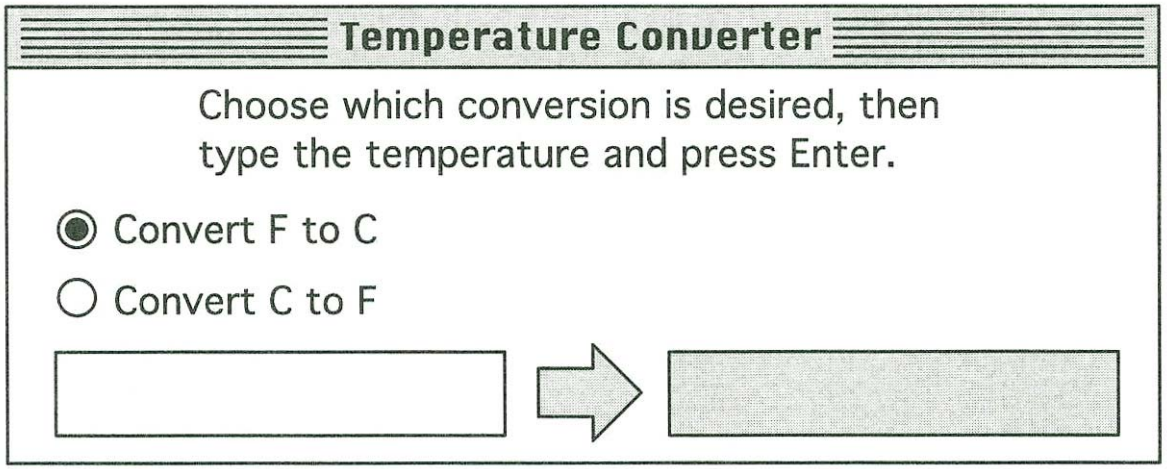
4b: if K terminate a variable string (parameter) keep the M in front of it

- *cd class↵: MKKKMKKKKKMK → MKKKMKKKKKMK*

Converting temperature: design 1

("Humane Interface", Raskin)

- Convert 92.5F to Celsius



Assume the focus is on the dialog box, so typing on the keyboard will enter text in the text field directly

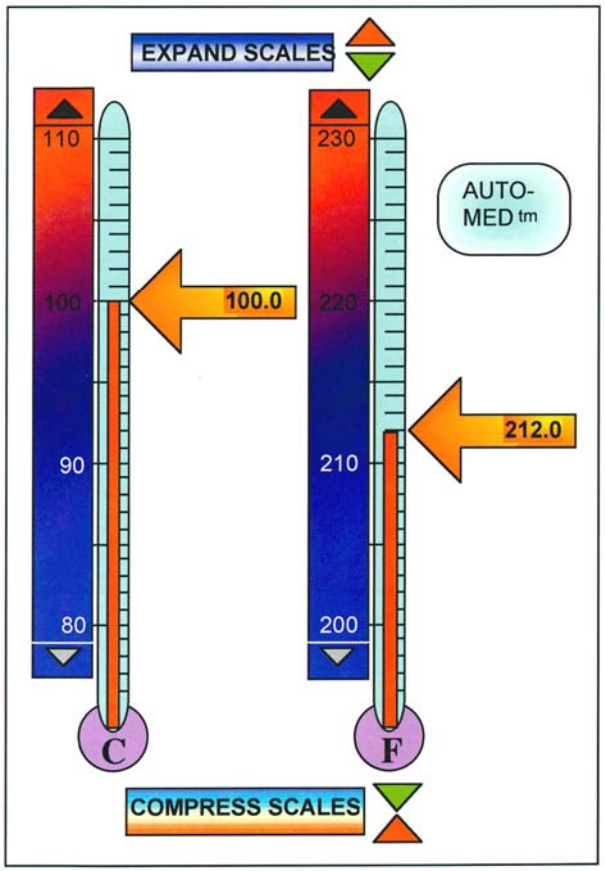
MKKKKMK (3.7s)

Average: 5.4s

HMPKHMKKKKMK (7.15s)

Converting temperature: design 2

(“Humane Interface”, Raskin)



HMPKPK (4.35s)

Average: 13.1s

HMPKSKMPKSKMPKSKMPKPK (21.9s)

Converting temperature: design 3

(“Humane Interface”, Raskin)

- Simple text interface with the following prompt:

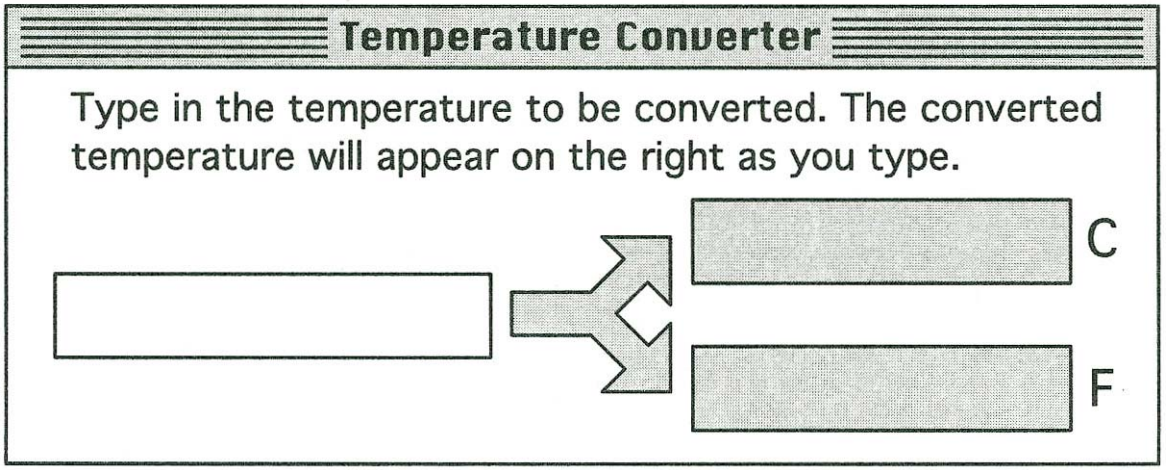
“To convert temperatures, type the numeric temperature, followed by C if it is in degrees Celsius or F it is in degrees Fahrenheit. The converted temperature will be displayed”

MKKKKMK (3.7s)

Average: 3.7s

Converting temperature: design 4

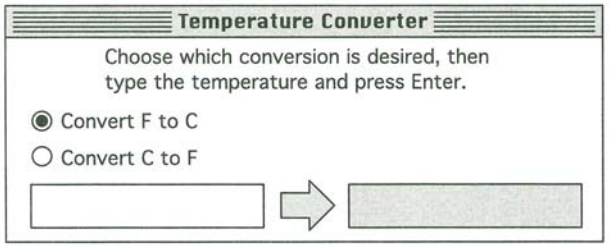
(“Humane Interface”, Raskin)

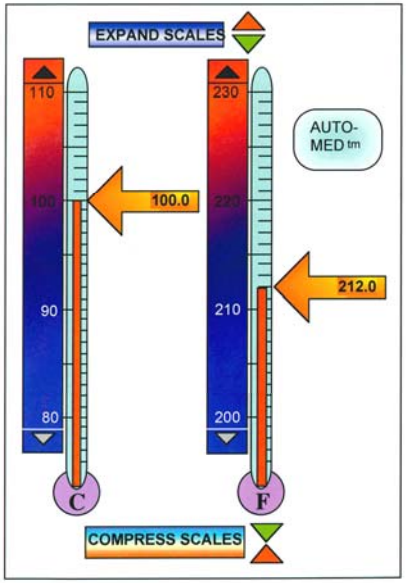


MKKKK (2.15s)

Average: 2.15s

Pros and Cons





"To convert temperatures, type the numeric temperature, followed by C if it is in degrees Celsius or F it is in degrees Fahrenheit. The converted temperature will be displayed"