CS109A Notes for Lecture 3/15/96

Representing Strings

Key issue: what operations will be performed on strings?

- Do the strings themselves change?
  - □ e.g., text editor storing lines of text.
- Does the set of strings change?
  - □ e.g., compiler maintaining the set of valid identifiers as it examines a program.
- What operations do we need to support?
  - □ Dictionary operations? Finding length? Etc.
- Do we have a limit on string length?
  - □ e.g., a personal record system that only stores 30 characters for names, addresses.

Storing Single Strings

Several approaches:

1. Character array with endmarker `\0` if there is a length limit.
2. Linked list of characters if the strings change length without bound.
   - □ But pack several characters per cell to save space, since pointers are 4–8 characters long.

Mass Storage of Strings

Often we need to store many strings in a large area (character array).

Example: Spelling checkers, searching text for many keywords.

Approaches:

1. String = pointer to place in storage array where word begins. Word terminated by marker, e.g., `\0`.

2. String = record with pointer to beginning of word in storage, plus length of word.

- Method (1) saves space, since it requires an extra character endmarker, vs. an integer length.

- However, method (2) might be better if we frequently needed to check the length of strings.

    - □ e.g., a Xword puzzle solver that searches for words of given length with 0 or more known letters.