

KWIC System

Hengbo Liu
YuWei Pai
Keith Nguyen
Xinhe Chen

Functional Requirements

- Components:
 - Has a web interface
 - Implements ADT architecture and functions
- Input
 - An ordered set of line
 - Each line is an ordered set of words
 - Each word is an ordered set of character
- Operation:
 - Repeatedly removing the first word and appending it at end of line.
- Output:
 - A listing of all circular shifts of all lines in ascending alphabetical order.

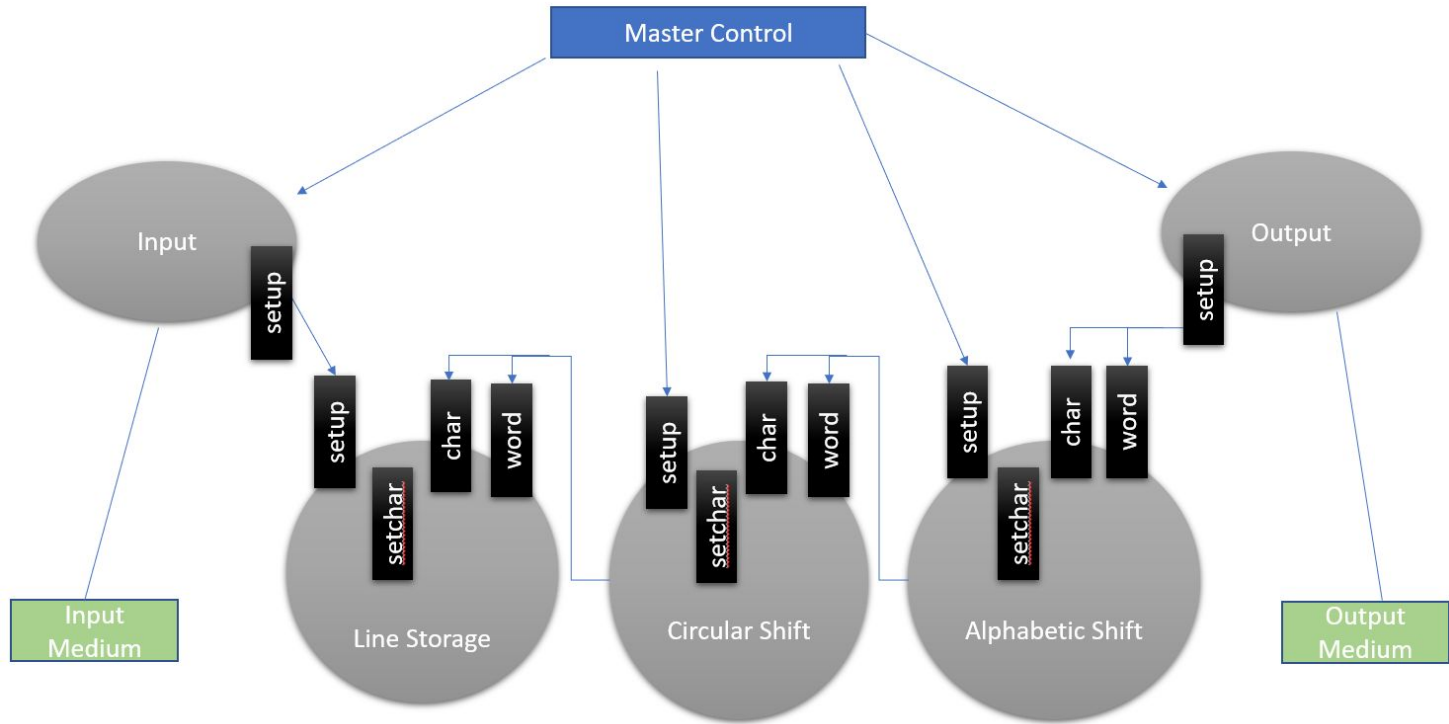
Non-functional requirement

- Good Performance
 - Fast response time
 - Minimal memory space
- Portability
 - platform independence- web browser
 - standard programming language
 - operating system
- Enhanceability
 - Noise word eliminator
- User-Friendly Access
 - Website interface
 - User manual
- Open Source
 - Open source code
- Reusability
 - Components as reusable entitles

Solutions

- Shared Data
 - Pros: Fast and efficient
 - Cons: Hard to modify/reuse
- ADT
 - Pros: Easy to reuse and modify
 - Cons: More time and memory consuming
- Implicit Invocation
 - Pros: Easy to reuse and modify
 - Cons: Difficult to control order of processing
- Pipe and Filter
 - Pros: Easy to reuse and modify
 - Cons: Hard to support interactive system (hard to enhance)

ADT



ADT

Data not shared

Each module provides interface

Components access data by invoking interfaces (information hiding)

Line Storage

setchar()

- used by Input module to store the char's that represent every character from the input.

char(), word()

- Used by Circular Shift module to retrieve all the indexes.

Circular Shift

setup()

- Uses char() and word() from the Line Storage to retrieve all the chars.
- Create the circular shifted sentences.

setchar()

- Uses the Circular shifted sentences and store each character as chars

char(), word()

- Used by Alphabetic Shift module to retrieve all the chars

Alphabetic Shift

setup()

- Uses char() and word() from the Circular Shift to retrieve all the circular shifted chars.
- Create the circular shifted sentences from the chars
- Sort the circular shifted sentences alphabetically

setchar()

- Uses the alphabetized sentences to store each character as chars

char(), word()

- Used by Output module to reconstruct the sentences from the input.

Demo

<http://www.utdallas.edu/~xxc170630/>