CMPE-013/L

Introduction to “C” Programming

Gabriel Hugh Elkaim
Winter 2015

Binary recursion
Binary recursion

- Code example:
  - TreePrint()
  - MorseEncode()
Software Engineering

Design process

- Initial Planning
- Requirements
- Analysis/Design
- Implementation
- Testing
- Evaluation
- Deployment

Software Engineering

Principles

- **Use consistent styling**
- **Summary:**
  - Utilize whitespace
  - *Good variable/function names*
  - Comments that describe non-obvious code behavior
    - "How?" and "why?" are good questions to answer in comments
Software Engineering
Formatting code

- Ugly code
- Beautiful code

Software Engineering
Formatting non-code

- Comments that describe non-obvious code behavior
  - "How?" and "why?" are good questions to answer in comments

```c
// First, determine the length of both items' data,
// given NULL data a -1 length so that it sorts to
// the head of the list.
int len1 = -1;
if (item1->data) {
    len1 = strlen(item1->data);
}
...```
Software Engineering

Principles

• Modularity is important

• Why?
  – Supports code reuse
  – Simplifies changes
  – Allows for testing

• How?
  – Keep functions small
  – Minimize side effects
  – Information hiding/encapsulation

Software Engineering

Principles

• Information hiding/encapsulation

• Summary:
  – Hide unimportant details from the user
  – Protects the user from breaking things
  – Separates backend from frontend
Software Engineering

Mantras

- Keep it simple, stupid
  - KISS
- Summary:
  - Don't solve problems you don't need to
  - Don't introduce unnecessary complexity
  - Prioritize for readability and modularity
  - Don't be clever and/or cute
  - Applies to code architecture and specific code constructs

Example

```c
ListItem *LinkedListGetFirst(ListItem *list)
{
    ListItem *tempPointer = NULL;
    if (list == NULL) {
        return NULL;
    }

    if (list->previousItem == NULL && list->nextItem != NULL) {
        return list;
    }
    else if (list->previousItem != NULL) {
        tempPointer = list;
        while (tempPointer->previousItem != NULL) {
            tempPointer = tempPointer->previousItem;
        }
    }
    return tempPointer;
}
```
ListItem *ListGetFirst(ListItem *list)
{
    while (list && list->previousItem) {
        list = list->previousItem;
    }
    return list;
}

Software Engineering

KISSL example

Software Engineering

Mantras

• Don't repeat yourself
  – DRY

• Summary:
  – Write code only once
  – Simplifies refactoring/incremental development
  – Avoids copy/paste errors
Software Engineering
Mantras

• You aren't gonna need it
  – YAGNI

• Summary:
  – Don't introduce features that are unnecessary
  – Don't write more code than you have to
  – Start small and build from there