CS 31 Discussion 1A, Week 5

Zengwen Yuan (zyuan [at] cs.ucla.edu)
Humanities A65, Friday 10:00—11:50 a.m.
Today’s focus

• Development skill: IDE debugging

• Project 3 hints
  • Function: reinforced
  • String: secret weapons

• Intro to arrays
IDE Debugging: concepts

• Debugging techniques
  • Breakpoints
  • Trace execution
    • single stepping, step into, step out, step over
  • Monitor variables and function calls
IDE Debugging: hands-on

http://web.cs.ucla.edu/classes/fall16/cs31/Codeexamples/phonenumber.txt
Project 3 hints: spec reading

• Read the spec (and the FAQ), again.

• What are the functions supposed to do?
  • What are the inputs/outputs?
  • What is the return type/value, what variables are changed?

• Can I provide some test cases (valid/invalid)

• Input/output compliance

• What should be included the report?
Project 3 hints: incremental development

- Logic, logic, logic: thinking effort is the key
- What are the major steps need to be done?
- What is the dependency of the functionalities?
- What is the simplest, base case for each function?
- How do I put the pieces together?
Project 3 hints: function call

- How to define a function?
- How to implement a function?
- How to call a function?
- What is call-by-reference and how to update reference parameters?
- Will your function return a value under every possible circumstance (if applicable)?
- How to get the return value (if applicable)?
- What is the difference between the names used in declaration, implementation, and real function call?
- (*)& function overloading and default arguments
Project 3 hints: string concepts

• Type matters:
  • string v.s. char
  • char v.s. integer

• Basic predefined functions:
  • isalpha(), isdigit(), isupper(),
    islower(), toupper(), tolower()
  • size(), substr()
Project 3 hints: string manipulation

• What is …?
  • slicing
  • concatenation

• How do I …?
  • convert string/char to integer
  • split a string into substrings
Project 3 hints: pseudocode

- A more effective means of communicating an algorithm
- Should NOT be merely a statement-by-statement rephrasing of the code
- Some basic constructs for flow of control are sufficient to implement any "proper" algorithm:
  - SEQUENCE is a linear progression where one task is performed sequentially after another.
  - WHILE, FOR, IF-THEN-ELSE, REPEAT-UNTIL, CASE
- Use indentation properly

http://www.unf.edu/~broggio/cop2221/2221pseu.htm
Project 3 hints: exercise
Intro to arrays

- An array is a series of elements of the same type placed in contiguous memory locations that can be individually referenced by adding an index to a unique identifier.

- In C++, indexing starts from 0.

- syntax:

  `<type>  <name>[size];`
Intro to arrays: declaration and initialization

```cpp
#include <iostream>
#include <string>
using namespace std;

int main() {

    // An array of called yarn that can hold 2 strings
    // uninitialized
    string yarn[2];

    // An array called foo that can hold 3 ints
    // initialized with values 0, 0, 0
    int foo [3] = {};

    // An array called bar that can hold 3 ints
    // initialized with values 1, 2, 3
    int bar [] = {1, 2, 3};
}
```