CSCI 2132
Software Development

Lecture 26:

Writing Large Programs

Instructor: Vlado Keselj

Faculty of Computer Science

Dalhousie University

Previous Lecture

- A common mistake with VLA declarations
- More string reading examples
- Buffer overflow risks
- String library functions

Command-Line Arguments

- To access the command-line arguments, we use another way to define the `main` function:

```
int main(int argc, char* argv[]) {
...
}
```

- `argc` is the number of command-line arguments
- `argv` is the array of command-line arguments; it is an array of pointers to `char`
- `argv[0]` is the program name as executed

Example: `sortwords` **program**

- Example: `sortwords` program, which sorts words given in the command line

- Usage example:

  `./sortwords orange apple banana`

- It should produce:

  ```
  apple
  banana
  orange
  ```

Example with argc and argv

- In the example:

  ```
  ./sortwords orange apple banana
  ```

- argc would be 4, and argv could be represented as follows

  ```
  argv
  argv[0] --------> ./sortwords\0
  argv[1] -------> orange\0
  argv[2] -------> apple\0
  argv[3] -------> banana\0
  argv[4] stores a NULL pointer
  ```

- Code (insertion sort):

  ```
  ~prof2132/public/sortwords.c
  ```

Writing Large Programs

- A large program consists of many modules
- Different programmers may work on different modules
- Logical to use one or more files for each module
  - Facilitates collaboration and reusing code
- Reading: C textbook: Chapter 15

Header Files

- Files that allow different source files (`*.c`) to share
  - Function prototypes
  - Type definitions
  - Macro definitions
  - etc.
- Naming convention: `*.h`

The `#include` **Directive**

- Tells the preprocessor to open a specified file and inserts its content into the current file

- Form 1: `#include <file_name>`
  - Search the directories in which system header files reside

  - On bluenose: `/usr/include,...`

- Form 2: `#include "file_name"`
  - First search the current directory, if not found then

  - directories in which system header files reside

- Question: Which form for your own header files?