EE1910
Coding Guide
(Dr. Johnson)

Last Updated 12/3/20

Coding Guide

- Windows Guidelines

  - Code Composer workspaces / projects will be placed in student directories – not inside the TI installation directory

  - Absolutely no spaces in file or directory names
    - use underscore instead: my_project_directory

  - File names must be descriptive
    - lab3_part1.c   or   continuous_average.c
    - l3p1, try15, i_hate_c
    - main.c is not an acceptable program name

Coding Guide

- General C code guidelines

  - All code must be properly indented
    - I will not help debug un-indented code
    - I will not grade un-indented code

  - Your name must be in the header section of your code

  - File structure will be:
        pre-processor directives
        function prototypes
        main
        function definitions

Coding Guide

- General C code guidelines

  - Main will be used for control

  - Most processing will be performed in functions
    - (after we learn about functions)

  - No global variables

  - Attempt to make your code as general (re-usable) as possible

  - Use #define for constants and values used in multiple locations

4

Coding Guide

- Comment guidelines
  - Header Example

File name

1 line description

Revision

Break up lines to prevent wrapping (hard to read)

General description

Technical description

Inputs and outputs

```c
/*
 * continuous_ave_no_functions.c
 *
 * Created on: Dec 7, 2018
 *     Author: johnsontimoj
 *
 * This program continuously averages inputs from the user
 *
 * In a continuous loop the user is asked for a new value
 * Each new value is then added to the running average of
 * all previous entries
 * The current average is then printed to the console
 *
 * The running average along with a count value is maintained
 * Each time a new value is entered the running average is
 * multiplied by the previous count, added to the new value
 * then divided by the new count
 *
 * This version does not use functions
 *
 * inputs - user input
 *          integer values to include in the average
 * output - console
 *          current average
 */
```

Coding Guide

- Comment guidelines
  - Main Example (no functions)

```c
#include <stdio.h>

int main(void){
    setbuf(stdout, NULL);    // code composer bug fix for console mode

    // variable declarations
    int input;
    int count;              // keep track of how many inputs we have received
    float running_ave;
    float tmp;              // calculation temporary variable

    count = 0;              // initialize variables
    running_ave = 0;

    // continuously request inputs and calculate average
    while(1){
        // ask for new input and increment the count
        printf("\nPlease enter an integer value: ");
        scanf("%i", &input);
        count = count + 1;

        // calculate the new average
        tmp = running_ave * (count - 1);    // current sum
        tmp = tmp + input;                  // new sum
        running_ave = tmp / count;

        //display the current ave
        printf("The current running average is: %f\n", running_ave);

    } // end while

    return 0;
} // end main
```

anywhere the variable name is not absolutely descriptive – provide a comment

Comment on what you are doing

comment on what this is
not
"add tmp and input"

as your code grows you will appreciate knowing what all those closing braces are for

Coding Guide

Main should be used for high level coordination.
(primarily does function calls)

- ## Comment guidelines
  - ### Main Example (with functions)

```c
#include <stdio.h>

int get_input(void);
float calculate_ave(float ave, int cnt, int input);

int main(void){
    setbuf(stdout, NULL);    // code composer bug fix for console mode

    // variable declarations
    int input;
    int count;              // keep track of how many inputs we have received
    float running_ave;

    count = 0;              // initialize variables
    running_ave = 0;

    // continuously request inputs and calculate average
    while(1){
        // ask for new input and increment the count
        input = get_input();
        count = count + 1;

        // calculate the new average
        running_ave = calculate_ave(running_ave, count, input);

        //display the current ave
        printf("The current running average is: %f\n", running_ave);

    } // end while

    return 0;
} // end main
```

anywhere the variable name is
not absolutely descriptive – provide
a comment

Comment on what you are doing

comment on what this is
not
"add tmp and input"

as your code grows you will appreciate
knowing what all those closing braces are for

Coding Guide

- Comment guidelines
  - Main Example (with functions)

```c
int get_input(void){
    // ask for new input and return the value
    int input;
    printf("\nPlease enter an integer value: ");
    scanf("%i", &input);

    return input;
} // end get_input

float calculate_ave(float ave, int cnt, int input){
    // calculate the new average
    float tmp;
    tmp = ave * (cnt - 1);    // current sum
    tmp = tmp + input;        // new sum
    ave = tmp / cnt;

    return ave;
} // end of calculate_ave
```

use descriptive function names

Comment on what you are doing

comment on what this is
not
"add tmp and input"