Welcome to COSC 1310

Introduction to Procedural Programming
Syllabus
History of C

• Invented by Dennis Ritchie in 1972 based on B (a simplified version of BCPL)
  • Ritchie was working for AT&T Bell Labs in Murray Hill, NJ
• Used to develop the Unix operating system and Unix commands
• Most system software such as OS are written in C or C++
• Replacement for assembly language for hardware interface.
• By late 1970's C had evolved to “K & R C“
History of C

• C Standards
  • 1\textsuperscript{st} standard created in 1989 by ANSI, ratified by ISO in 1990. It is called C89, but some call it C90.
  • 2\textsuperscript{nd} standard was ratified in 1999, called C99.
    Numerical extensions such as complex numbers, variable length arrays, and IEEE floating-point arithmetic are major enhancements.
  • 3\textsuperscript{rd} standard is C11, which most notably has introduced support for multi-threading.

http://www.iso-9899.info/wiki/The_Standard
Lesson 1

FIGURE 1.2
The steps involved in C program development
QUIZ

What are the 4(5) steps in the development of a C program?
QUIZ

What are the 4(5) steps in the development of a C program?

• Edit
• (Preprocess)
• Compile
• Link
• Execute
Lesson 1

What is an IDE?

• MS Visual Studio

• IDE One → www.ideone.com
Lesson 1

Let’s start!

The “Hello, World!” program
#include <stdio.h>

int main(void) {
    printf("Hello, World!\n");
    return 0;
}

To do for next time:

• Read the entire Lesson 1 and take notes in the notebook.
• Answer Quiz questions 1-10 in the notebook.
• Solve Exercises 2-6 (in MSVC or a web compiler like IDE One).
QUIZ

What is an IDE?
QUIZ

What are the 4(5) steps in the development of a C program?
QUIZ

What are the 4(5) steps in the development of a C program?

• Edit
• (Preprocess)
• Compile
• Link
• Execute
Coding style
What is different?

```c
#include <stdio.h>

int main(void)
{
    printf("Hello, World!\n");
    return 0;
}
```

```c
#include <stdio.h>

int main(void)
{
    printf("Hello, World!\n");
    return 0;
}
```
Bug buster #4

```c
#include <stdio.h>

int main(void) {
    printf("Hello, World!\n");
    return 0;
}
```
Bug buster #5

```c
#include <stdio.h>

int main(void)
{
    printf("Hello, World!\n");
    return 0;
}
```
Lesson 2

The elements of a C program
```c
#include <stdio.h>
#define TARGET_AGE 88

int year1, year2;

int calcYear(int year1);

int main(void){
    printf("What year was the subject born? ");
    scanf(" %d", &year1);

    year2 = calcYear(year1);

    printf("%d blah %d blah %d", year1, TARGET_AGE, year2);
    return 0;
}
```
return 0;
}

int calcYear(int year1){
    return (year1 + TARGET_AGE);
}
A larger example (not in text)

```c
/* Sorts an array using Bubble-Sort 
   Array initialized w/random integers */
#include <stdio.h>
#include <stdlib.h>    // needed for rand()
#define SIZE 10

void printArray(const int b[], int size);

int main() {
  int a[ SIZE ];
  int pass; // passes counter
  int i;   // comparisons counter
  int temp; // temporary location used to swap array elements

  for(i=0; i<SIZE; i++)
    a[i] = rand()%100;  // random integers in what range?
```
for ( pass = 1; pass < SIZE; pass++ ) {
    for ( i = SIZE-2; i >=0; i-- ) {
        /* compare adjacent elements, swap if not in order */
        if ( a[ i ] > a[ i + 1 ] ) {
            temp = a[ i ];
            a[ i ] = a[ i + 1 ];
            a[ i + 1 ] = temp;
        }
    }
    printArray(a, SIZE); //comment this out if large array
}

printf( "\n\nData items in ascending order:" );
printArray(a, SIZE); /* output sorted array */
printf( "\n" );
/* end main */

void printArray(const int b[]){
    int i;
    printf("\n");
    printf("\n");
    for ( i = 0; i < SIZE; i++ ) {
        printf( "%6d", b[ i ] );
    }
}
```c
{ 

    printf("%d\n", b[i]);
    for i = 0; i < SIZE; i++)
        printf("\n");
}

void printarray(const int p[] b)
{

    /* end main */
    printf("\n");
    /* output sorted array */
    printf("%d\n", a[0]);
    printf("%d\n", a[1]);
    printf("%d\n", a[2]);
    printf("%d\n", a[3]);
    printf("%d\n", a[4]);
    } ( [ i + 1 ] < [ i ] )
    } ( [ i ++ ] > ssed )

    // comment this out if large array
    printf("%d\n", SIZE); /*
    {
    temp = [ i + 1 ]
    [ i + 1 ] = [ i ]
    [ i ] = temp
    */
    // compare adjacent elements, swap it not in order
    } ( - = i )
    for ( -- i = SIZE-2; i = i ; i++)

    printf("%d\n", [ i ] e)
    printf("%d\n", [ i ] e)
    printf("%d\n", [ i ] e)
    printf("%d\n", [ i ] e)
    printf("%d\n", [ i ] e)

    // random integers in what range?
    } ( i ++ )
    for ( i = 0; i < SIZE; i++)

    int temp; temporary location used to swap array elements
    int i; comparisons counter
    int j; pass2 counter
    int a [ SIZE ];
}

#define SIZE 10

() // needed for rand()
#include <stdlib.h>
#include <stdio.h>

/* Array sorts an array using bubble-sort */
```
To do for next time:

• Read Lesson 2, and specifically their larger example in Listing 2.2. Take notes in the notebook.
• Answer Quiz questions 1-10 in the notebook.
• Solve coding Exercises 1, 2 in an IDE.
Homework for Chapters 1 and 2:
• Ch.1 – Exercise 6
• Ch.2 – Exercises 4 and 5

For each exercise:
• Capture a screenshot of both code and output
• Answer in your own words the question, under the screenshot

Due Wed, Sept.7, at beginning of class.

Use 65 instead of 1 in printf()

Line 9 in Ex. 5 is incomplete. It should read:
fgets(buffer, 42, stdin);