Clase #03 de 27

Introducción a Léxico, Sintaxis, Semántica y Pragmática

Abril 8, Martes

Agenda para esta clase

- Introducción al Lenguaje de Programación C
- “El Libro Blanco”
- Análisis y Síntesis de Hello.c
- Otras versiones de Hello.c
- Introducción a Léxico, Sintaxis, Semántica, y Pragmática

Introducción al Lenguaje de Programación C

Descripción general

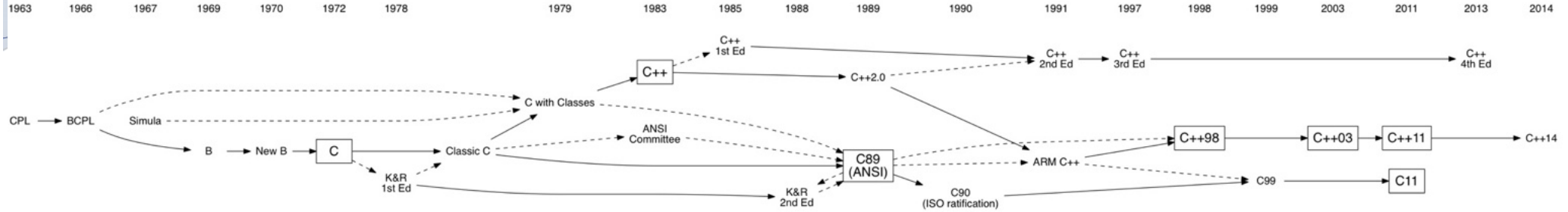
- LP de propósito general, no está especializado
- Economía en las expresiones, pero expresivo
  - Poco texto, mucha información
  - Variabilidad en texto, variabilidad en significado.
- Control de flujo (¿de qué?)
- Estructuras de datos
- Gran cantidad de operadores
- Flexible
- No es de muy alto nivel (¿de qué?)
- No es grande (¿en qué sentido?)
- Su falta de restricciones y su generalidad lo hacen efectivo
- Independiente de máquina, portable (procesador y sistema operativo)
- El lenguaje de programación de Unix

- Primer lenguaje de alto nivel eficiente y portable
  - En el momento, menos problemas que
    - Basic, PL/I, Fotran, Cobol, Pascal
  - Comparado con Lisp
    - Vinculación
    - Más rápido
    - Con GC, lo cual no es apropiado para programación de sistemas
  - C es la mejor abstracción de una computadora existente, no de un dispositivo imaginario
  - Suficientes estructuras de control y de datos para resolver problemas, limitadas para que se pueda implementar el compilador.

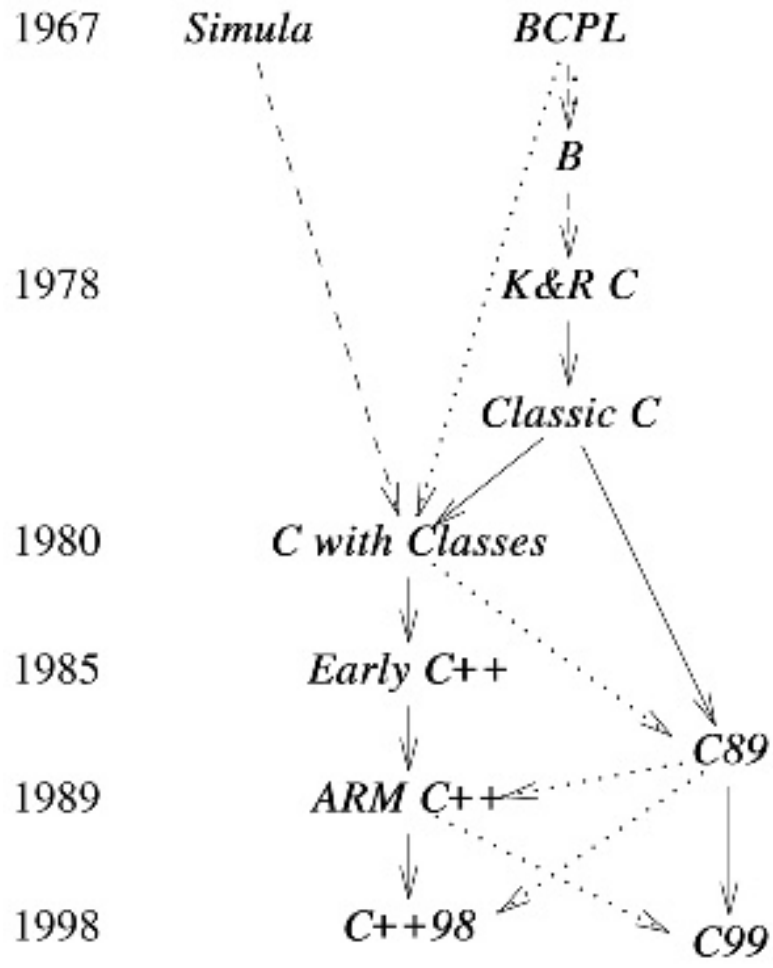
Frases sobre C

- C no es un LP grande, y no le queda bien un libro grande (K&R)
- C tiene vueltas, falencias y un enorme éxito (Ritchie)
- C es un arma filosa, con la se pueden hacer programas eficientes y elegantes o una "carnicería" (Pike)
- C mejora a medida que uno gana experiencia con C (K&R) (Curva de aprendizaje empinada).

Historia de C y LP relacionados



- 1969-1973
  - Elaboración. Basado en B, a su vez, basado en BCPL
  - C Pre estandarización
- 1978
  - K&R 1era edición
- 1983
  - Comienza estandarización
- 1988
  - K&R 2da edición
- 1989-90
  - C89–C90 (1era versión estándar)
- 1999
  - C99 (2da versión estándar)
- 2011
  - C11 (3era versión estándar)
- 2019
  - C18 (4ta versión, sin cambios visibles, también se lo conoce como C17)
- 2023
  - C23 (publicado en 2024)

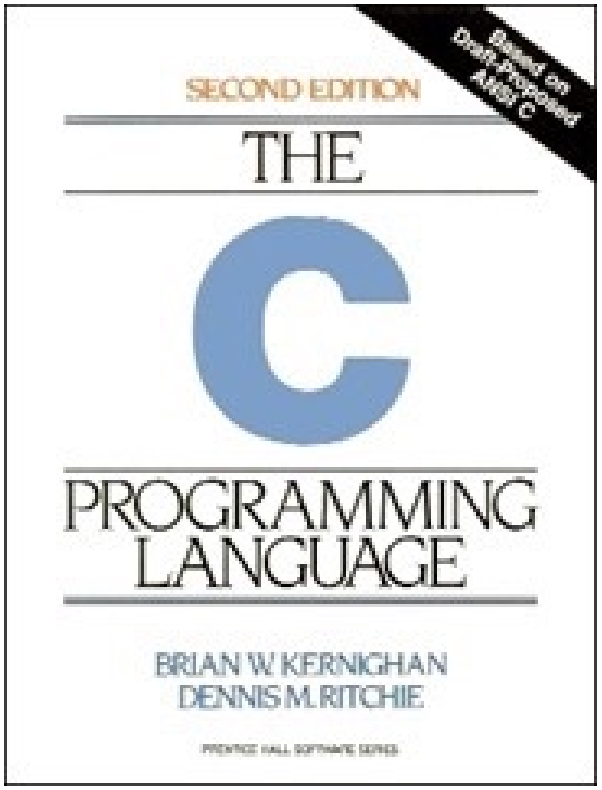


- Otros LP
  - C++
  - Objective-C
  - C#
  - D
  - Java

El Libro Blanco

K&R 1.1 Una Introducción Tipo Tutorial – Empezando

“El Lenguaje de Programación C” aka “El Libro Blanco” aka “K&R”



- Autores:
  - Dennis Ritchie autor del LP y coautor de Unix con Ken Thompson
  - Kernighan
- Requisitos
  - Conocimiento de programación
  - Lectura atenta
- Ejercicios
- Ediciones y usos
  - 1978 1era Edición
    - Manual de referencia
  - 1988 2da Edición
    - ANSI C
- Preliminares (Front Matter)
  - Prefacio
  - Prefacio de la primera edición
  - Introducción
- General, “Ancho”
  - Capítulo 1: Tutorial
- En “profundidad”: Capítulos 2 a 7
  - 2 Tipos, Operadores y Expresiones
  - 3 Control de Flujo
  - 4 Funciones y Estructura de Programa
  - 5 Punteros y Arreglos
  - 6 Estructuras
  - 7 Entrada y Salida
- Ejemplo Integrador
  - 8 Interfaz de Sistema de UNIX – entrada/salida, sistema de archivos y alocación de memoria
- “Apéndices”
  - A: Especificación del LP: Semántica (LN) y Sintaxis (BNF)
  - B: Biblioteca estándar
  - C: Cambios introducidos en la primera versión del estándar.

"Hello, World!"– Demostración de Compilación y Ejecución

```
#include <stdio.h>
int main() {
  printf("Hello, World!\n");
}
```

```
$ cc hello.c

$ ./hello
Hello, World!
```

Ejercicios

- 1-1. Experimentar con eliminación de partes y compilar
- 1-2. Probar diferentes \c.

Análisis y Síntesis de Hello.c

Análisis de 'Hello, World!'

```
#include <stdio.h>
```

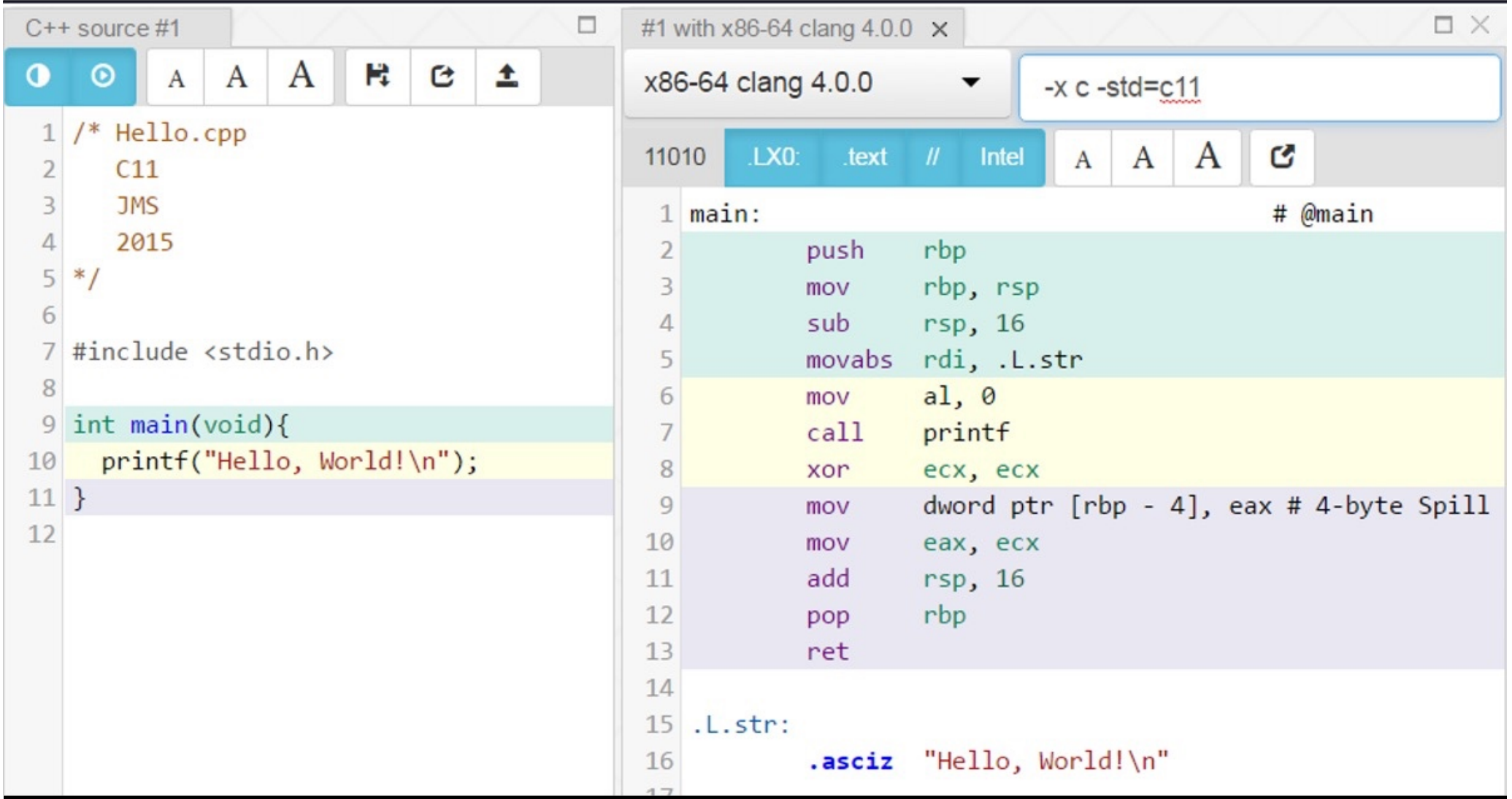
Incluye información acerca de la biblioteca estándar

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

Define una función llamada *main* que no recibe valores argumento. Las sentencias de *main* se encierran entre llaves

main llama a la función de biblioteca estándar *printf* para imprimir esa secuencia de caracteres. \n representa el carácter nueva-linea

- Estructura de un programa
  - vs. Pascal
    - Lineal vs. Jerárquico
- Rol de main
- Preprocesador
- Biblioteca estándar: Entrada/Salida
  - vs Framework
- Rol de llaves { }
- Función printf
- Pasaje de argumentos a funciones
- Constantes de cadena o literal cadena
- Secuencia de escape
- Punto y coma como terminador
  - vs. Pascal
- Analizar que es "palabra" del LP y que no
- Identificadores: main y printf -- ¿Cuál es el autómatafinito que los reconoce?



- www.CompilerExplorer.com

Generalización de hello.c

Programa General

- Un programa es una secuencia de funciones. Forma general:
  ```
  main
  f
  g
  …
  ```
- Forma general de una función
  ```
  Tipo Nombre(Parámetros){
  ```
- ```
      Cuerpo
  }
  ```
- Cuando se corre (ejecuta) un programa, por convención, main es la primera función invocada por el ambiente de ejecución (e.g., sistema operativo). Todo programa debe tener un main con o sin parámetros

hello.c

- Este programa define solo la función main
- Como toda función, main puede tener o no parámetros
- Este main invoca a printf con una cadena literal (cadena constante) como argumento. "Entre comillas".
  ```

Otras Versiones de Hello.c

Diferencias con Estándar C (y con C++)

```
main( ){
  puts("Pre Ansi");
}
```

```
#include <stdio.h>

int main(void){
  puts("Post Ansi");
  return 0;
}
```

```
#include <stdio.h>
#include <stdlib.h>

int main(void){
  puts("Post Ansi");
  return EXIT_SUCCESS;
}
```

```
#include <stdio.h>

int main(){
  puts("Post Ansi");
}
```

```
#include <iostream>

int main() {
  std::cout << "ANSI C++\n";
}
```

Introducción a Léxico, Sintaxis, Semántica, y Pragmática

Otras versiones – ¿Mismos Léxico, Sintaxis, Semántica, y Pragmática?

```
#include <iostream>

int main() {
  std::cout << ”Hello, World\n";
}
```

```
#include <stdio.h>
int main() {
 printf( "Hello, World!\n" );
}
```

```
#include <stdio.h>
int main(){
 printf( "Hello,"
         "World!"
         "\n"     );
}
```

```
#include <stdio.h>
int main(){
 printf( "Hello,"  );
 printf( " World!" );
 printf( "\n"      );
}
```

```
#include <stdio.h>
int main(){
 puts( "Hello, World!" );
}
```

Términos de la clase #03

Definir cada término con la bibliografía

- Análisis y Síntesis de hello.c
  - main
  - printf
  - Función del #include
  - puts
  - Secuencia de escape
  - Terminador de sentencia de C
  - Separador de sentencias de Pascal
  - Archivo .h (Encabezado)
  - String literal: Cadena Literal, o Literal de cadena, o Constante cadena, o Cadena constante
  - Lenguaje Ensamblador
  - Stack
  - sp
  - bp
  - call
  - Estructura lineal de un programa C
  - Estructura jerárquica de un programa Pascal
- Otras versiones de Hello.c
  - Tipo int implícito
  - Valor retornado por main
  - Concatenación de cadenas
  - Múltiples invocaciones
  - Operación corrimiento en C y en C++: <<
  - Operación inserción en C++ <<
  - cout
  - stdout
  - printf versus puts
  - printf versus fprintf
  - puts
  - EXIT_FAILURE
  - stdlib.h
- Introducción a:
  - Léxico
  - Sintaxis
  - Semántica
  - Pragmática

Tareas para la próxima clase

1. Leer de K&R1988:
   - A.2],A.10], A.11.2, A.12.
2. Responder:
   - ¿Que son los *trigraph*? ¿Por qué se agregaron? ¿Por qué se sacaron? ¿Como lo relaciona con https://imgs.xkcd.com/comics/workflow.png? ¿Cómo lo relaciona con Semantic Versioning?

¿Consultas?

Fin de la clase