

C+ Basic

王淇 (LittleCube)

台南一中資訊社 TFCIS

2021.9.22

1 編譯

- 1 基本概念

- 2 參數

- 3 預處理器

2 錯誤

3 Undefined Behavior

- `.cpp` → `.exe`/可執行檔

- `g++ example.cpp → a.out`

- `g++ example.cpp → a.out`
- `g++ -o example.exe example.cpp → example.exe`

- `g++ example.cpp → a.out`
- `g++ -o example.exe example.cpp → example.exe`
- `-std=gnu++14`

- `g++ example.cpp → a.out`
- `g++ -o example.exe example.cpp → example.exe`
- `-std=gnu++14`
- `-Wall`

- `g++ example.cpp → a.out`
- `g++ -o example.exe example.cpp → example.exe`
- `-std=gnu++14`
- `-Wall`
- `-O1, -O2, -O3, -Ofast`

讓編譯器在編譯的時候幫你處理好的東西

讓編譯器在編譯的時候幫你處理好的東西

- Replacing text macros
- Source file inclusion
- Conditional inclusion
- Implementation defined behavior control

- `#define identifier replacement-list`

- `#define identifier replacement-list`
- `#define identifier(parameters) replacement-list`

- `#define identifier replacement-list`
- `#define identifier(parameters) replacement-list`
- `#define identifier(parameters, ...)
replacement-list`

- `#define identifier replacement-list`
- `#define identifier(parameters) replacement-list`
- `#define identifier(parameters, ...)
replacement-list`
- `#define identifier(...) replacement-list`

- `#define identifier replacement-list`
- `#define identifier(parameters) replacement-list`
- `#define identifier(parameters, ...)
replacement-list`
- `#define identifier(...) replacement-list`
- `#undef identifier`

- `#include <filename>`
- `#include "filename"`

- `#if` expression

- `#if` expression
- `#ifdef` identifier

- `#if` expression
- `#ifdef` identifier
- `#ifndef` identifier

- `#if` expression
- `#ifdef` identifier
- `#ifndef` identifier
- `#elif` expression

- `#if` expression
- `#ifdef` identifier
- `#ifndef` identifier
- `#elif` expression
- `#else`

- `#if` expression
- `#ifdef` identifier
- `#ifndef` identifier
- `#elif` expression
- `#else`
- `#endif`

編譯 - 預處理器 - Implementation defined behavior control

- `#pragma pragma_params`

編譯 - 預處理器 - Implementation defined behavior control

- `#pragma pragma_params`
- <https://gcc.gnu.org/onlinedocs/gcc/Pragmas.html>

1 Syntax Error/Compile Error

- 1 Syntax Error/Compile Error
 - 語法錯誤，通常搜尋一下就有了

- 1 Syntax Error/Compile Error
 - 語法錯誤，通常搜尋一下就有了
- 2 Runtime Error

- 1 Syntax Error/Compile Error
 - 語法錯誤，通常搜尋一下就有了
- 2 Runtime Error
 - 執行時產生的錯誤

1 Syntax Error/Compile Error

- 語法錯誤，通常搜尋一下就有了

2 Runtime Error

- 執行時產生的錯誤
- 溢位、戳到不該戳的記憶體、`std::bad_alloc`、.....

1 Syntax Error/Compile Error

- 語法錯誤，通常搜尋一下就有了

2 Runtime Error

- 執行時產生的錯誤
- 溢位、戳到不該戳的記憶體、`std::bad_alloc`、.....
- **不是有錯誤就一定會報錯**

1 Syntax Error/Compile Error

- 語法錯誤，通常搜尋一下就有了

2 Runtime Error

- 執行時產生的錯誤
- 溢位、戳到不該戳的記憶體、`std::bad_alloc`、.....
- **不是**有錯誤就一定會報錯

3 Segment Fault

錯誤

1 Syntax Error/Compile Error

- 語法錯誤，通常搜尋一下就有了

2 Runtime Error

- 執行時產生的錯誤
- 溢位、戳到不該戳的記憶體、`std::bad_alloc`、.....
- **不是**有錯誤就一定會報錯

3 Segment Fault

- 記憶體產生的錯誤

1 Syntax Error/Compile Error

- 語法錯誤，通常搜尋一下就有了

2 Runtime Error

- 執行時產生的錯誤
- 溢位、戳到不該戳的記憶體、`std::bad_alloc`、.....
- **不是**有錯誤就一定會報錯

3 Segment Fault

- 記憶體產生的錯誤
- 試圖存取不存在、沒有權限、或唯讀的記憶體

1 Syntax Error/Compile Error

- 語法錯誤，通常搜尋一下就有了

2 Runtime Error

- 執行時產生的錯誤
- 溢位、戳到不該戳的記憶體、`std::bad_alloc`、.....
- **不是**有錯誤就一定會報錯

3 Segment Fault

- 記憶體產生的錯誤
- 試圖存取不存在、沒有權限、或唯讀的記憶體
- 很有可能是由記憶體超出限制或指標維護不良所產生

1 Syntax Error/Compile Error

- 語法錯誤，通常搜尋一下就有了

2 Runtime Error

- 執行時產生的錯誤
- 溢位、戳到不該戳的記憶體、`std::bad_alloc`、.....
- **不是**有錯誤就一定會報錯

3 Segment Fault

- 記憶體產生的錯誤
- 試圖存取不存在、沒有權限、或唯讀的記憶體
- 很有可能是由記憶體超出限制或指標維護不良所產生
- 注意這個跟 Runtime Error **有相重疊**的地方，取決於哪個先發生

1 Syntax Error/Compile Error

- 語法錯誤，通常搜尋一下就有了

2 Runtime Error

- 執行時產生的錯誤
- 溢位、戳到不該戳的記憶體、`std::bad_alloc`、.....
- **不是**有錯誤就一定會報錯

3 Segment Fault

- 記憶體產生的錯誤
- 試圖存取不存在、沒有權限、或唯讀的記憶體
- 很有可能是由記憶體超出限制或指標維護不良所產生
- 注意這個跟 Runtime Error **有相重疊**的地方，取決於哪個先發生

4 Illegal instruction

錯誤

1 Syntax Error/Compile Error

- 語法錯誤，通常搜尋一下就有了

2 Runtime Error

- 執行時產生的錯誤
- 溢位、戳到不該戳的記憶體、`std::bad_alloc`、.....
- **不是**有錯誤就一定會報錯

3 Segment Fault

- 記憶體產生的錯誤
- 試圖存取不存在、沒有權限、或唯讀的記憶體
- 很有可能是由記憶體超出限制或指標維護不良所產生
- 注意這個跟 Runtime Error **有相重疊**的地方，取決於哪個先發生

4 Illegal instruction

- 編譯出來的程式含有無法執行的部分，通常是你的程式寫爛了