

Preprocesszor


Pataki Norbert



Programozási Nyelvek és
Fordítóprogramok Tanszék

Programozási Nyelvek (C++)

Szankciós infláció



ICS 35 > 35.060

ISO/IEC 14882:2020

Programming languages — C++

ABSTRACT [Preview](#)

This document specifies requirements for implementations of the C++ programming language. The first such requirement is that they implement the language, so this document also defines C++. Other requirements and relaxations of the first requirement appear at various places within this document.

BUY THIS STANDARD

FORMAT	LANGUAGE
✓ PDF	English

CHF **198** [BUY](#)

ISO/IEC 14882:2020

Programming languages — C++

Abstract

[Preview](#)

This document specifies requirements for implementations of the C++ programming language. The first such requirement is that they implement the language, so this document also defines C++. Other requirements

and relaxations of the first requirement appear at various places within this document.

C++ is a general purpose programming language based on the C programming language as described in

ISO/IEC 9899:2018 Programming languages — C (hereinafter referred to as the C standard). C++ provides

Buy this standard

Format	Language
✓ PDF	English

CHF **208** [Buy](#)

Bevezetés

- A processzor csak a gépi kódot érti, pl. C/C++ kódot nem tud egyből végrehajtani.
- Futtatás előtt a C/C++ programokat le kell fordítani.
- Fordítóprogramot használunk (pl. g++).
- Fordítóprogramok a programok bizonyos tulajdonságait ellenőrzik, ez növeli a program helyességének esélyét, de nem garantálja azt.
- A fordítóprogramok optimalizál(hat)ják a kódot.

Bevezetés



Lépések

- 1 Preprocesszási fázis (preprocessor)
- 2 Nyelvi fordítás (compiler)
- 3 Összeszerkesztés (linker)

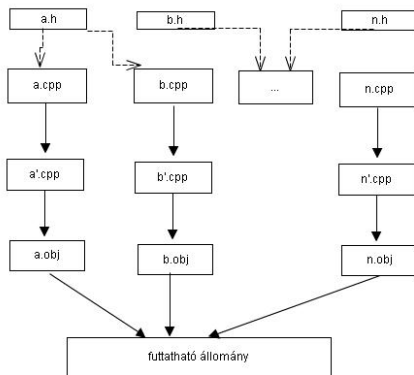
Forrás file-ok

- Fordítási egység: .cpp
- Header file-ok

Modularizálás

- Egy programot általában több (sok) file-ban írunk meg, *modularizáljuk*
- Csapatmunka növelése
- Karbantarthatóság
- Csak a szükséges file-okat fordítsuk újra (`make`, `Makefile`)

Fordítás folyamata



Alapok

- Szövegátalakító eszköz
- File tartalmának beillesztése
- Szövegrészek kivágása/benntartása
- Search & replace
- Nyelv-független eszköz, nem foglalkozik azzal, hogy C/C++ kódot dolgoz-e fel
- Csak a szükséges esetekben használjuk
- Automatizálás bizonyos esetekben
- C++ nyelvi konstrukciók
- Azonosítók konvenciója

Direktívák

- #
- Szövegátalakítás, nem végrehajtás
- *Direktívák*, nem utasítások

Alapvető direktívák

- `#include`
- `#define`, `#undef`
- `#ifdef`, `#ifndef`,
- `#if`, `#else`, `#elif`, `#endif`

Példák

```
// a.cpp:
void f( int i )
{
    // ...
}
```

```
// b.cpp:
void g( int i )
{
    f( i );
}
```

```
b.cpp: In function 'void g(int)':
b.cpp:3:8: error: 'f' was not declared in this scope
    f( i );
    ^
```

Példák

```
// a.cpp:
void f( int i )
{
    // ...
}

// b.cpp:
void f( int );

void g( int i )
{
    f( i );
}
```

Lefordul

Header file-ok

- Sok fordítási egység
- Sok függvény
- Karbantartás
- Automatizálás
- Függvények deklarációi header file-okban
- Header file-ok automatikus bemásolása preprocesszorral

Példák

```
// a.cpp:  
void f( int i )  
{  
    // ...  
}
```

```
// a.h:  
void f( int );
```

```
// b.cpp:  
#include "a.h"  
  
void g( int i )  
{  
    f( i );  
}
```

Tömb méretének paraméterezése

```
#define N 20
#define M 30

...
int x[ N ];
int y[ N ];
int z[ M ];

for( int i = 0; i < N; ++i )
{
    y[ i ] = x[ i ];
}
```


Makrók

```
#define SQUARE( x ) x * x  
// ...  
std::cout << SQUARE( 5 );
```

Makrók

```
#define SQUARE( x ) x * x  
// ...  
std::cout << SQUARE( 2 + 3 );
```

Makrók

```
#define SQUARE( x ) ((x) * (x))  
// ...  
std::cout << SQUARE( 2 + 3 );
```

Makrók

```
#define SQUARE( x ) ((x) * (x))  
// ...  
int i = 5;  
std::cout << SQUARE( i++ );
```

Include guard

- Osztályok megosztása fordítási egységek között
- Include guard
- Pragma once

Példa

```
// complex.h:
#ifndef COMPLEX__H
#define COMPLEX__H

class Complex
{
private:
    double re, im;

public:
    double abs();
    // ...
};

#endif
```

Feltételes fordítás – demo

```
g++ -dM -E -x c++ - < /dev/null
```

C/C++ feltételes fordítás

```
#ifdef __cplusplus
extern "C" {
#endif
// ...
#ifdef __cplusplus
}
#endif
```


Fordítás, futtatás

- Interpretált nyelvek
- Byte code
- JIT compiler

