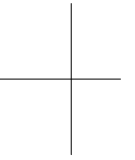
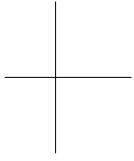


C++

C++ 가 , C++가 C , C 가
 가 ,
 C++
 . C++
 ,
 C++ 가 , C++ 가
 , C++
 . C++
 가
 C++
 C 가 ,
 가 ,
 가





C++

,

.

,

.

,

,

.

.

,

가,

.

가

, C++

가

.

,

C++

.

.

,

,

.

,

.

,

,

,

.

(abstraction)

,

가

,

.

,

.

,

,

.

,

가

.

,

,

,

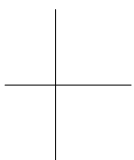
.

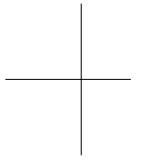
,

.

,

.





가 , C++가 C++

C++ 가

C++

Stroustrup The C++ Programming Language (Addison-Wesley, 2000) C++

C++

C++

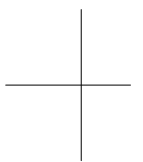
가 , C++ C

가 . 3

6

6

C++ 가



! GUI

C C++

C++ C

C++ C

가 C++

(

), C++ C

C++ C

C++

10 printf malloc

1 string

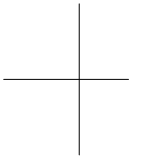
!

7

8

. C++

가



C++

가

가

C++

() ‘ 가 가

가

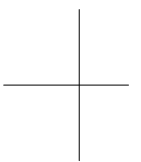
A, B

가

C++ , ht tp : : / / www . acce l er at edc p p . co m

. C++

가



가

Robert Berger, Dag Bruck, Adam Buchsbaum, Stephen Clamage, John Kalb, Jeffrey Oldham,
David Slayton, Bjarne Stroustrup, Albert Tenbusch, Bruce Tetelman, Clovis Tondo .

Addison-Wesley ,

Tyrrell Albaugh, Bunny Ames, Mike Hendrickson, Deborah Lafferty, Cathy Ohala, Simone

Payment . Alexander Tsiris 13.2.2 .

가

가



Chapter 0

```
...  
, C++  
//  
#include <iostream>  
  
int main()  
{  
    std::cout << "Hello, world!" << std::endl;  
    return 0;  
}
```

```

        (Hello, world!)
    ,
    .
    Hello, world!
    (standard output)
    ,
    가
    C++ , http://www.accelerated
    cpp.com
    ,
    가
    , (implementation)1)
    , C++

```

0.1

```

    가
    // C++
    // (comment)
    ,

```

0.2 #include

```

    C++ ,
    (standard library)
    , (Core language)가
    ,

```

¹⁾ C++ , Microsoft Visual C++ , GNU GCC

C++ ,
 .
 #include (directives)
 .
 .
 #include <iostream>
 ostream . . . ,
 . . . ,
 ostream #include , (angled bracket, "<" ">")
 , (standard header) C++ .
 C++ 가 ,
 . (include) 가

0.3 main

(function) 가 , (call) ,
 . C++ main ,
 . 가 C++ ,
 .
 main (integer) ,
 . (0) ,
 가 , int main
 .
 int main()
 . , int (core language)가
 . main 가 (parameters)

. main , 가 , .
10.4 .

0.4

(*curly braces*) (statements,) main

```
int main()
{
    //
    // (statement)
}
```

C++ , .
main , , .
 , , .

0.5

```
std::cout << "Hello, world!" << std::endl;
```

(*output operator*), <<
Hello, world! , std::endl .
std::가 , std (namespace)
 , 가 .
std , iostream
cout endl , std::cout
std::endl .

`std::cout` (*standard output stream*), C++
 가 `std::cout`
 가 `std::cout`
`std::endl`

0.6 return

`return`,
`return 0;`
`return` (;) (
 0) 가
`main` `int`, `main`
 C++ `main` `return`
 ,
`return` 가
`return`

0.7

C++ 가 가
 (expression) (scope)
 , 가
 (side effect)
 (result)가
 , 3+4 7
`std::cout << "Hello, world!" << std::endl`

Hello, world!

(operator) (operand) , 가
 Hello, world! , << std::cout,
 "Hello, world!" std::endl .
 (type) . ,
 , (operation) .
 가 .
 가 , int
 , std::ostream .
 , std::cout std::ostream 가 .
 << 가 . , <<
 ? << (left
 associative) . , <<가
 , << 가
 "Hello, world!" , std::cout 가 . <<
 std::endl , std::cout << "Hello, world!" 가 .
 ,
 .
 (std::cout << "Hello, world!") << std::endl
 << . <<
 std::cout 가 . , std::cout std::ostream .
 (literal) . 10.2 .
 , << 가
 . , 가 .
 << std::ostream std::cout .
 std::endl , (manipulator) .

```

    . <<
    가 std::ostream
    ,
    가
    , <<
    . std::endl
    ,
    .
    ,
    std::cout
    ,
    Hello, world!
    ,
    ,
    가
    .
    .
    .
    (scope)
    .
C++
    ,
    .
    가
    .
    .
    std
    .
    ,
    .
    ,
    std
    .
    ,
    ,
    std::cout
    cout
    std
    .
    .
    std::cout
    (qualified name)
    , ::
    .
    (scope operator)
    . ::
    (
    )
    .
    . std::cout
    std
    . ::
    .
    , std::cout
    "cout
    std
    (
    )
    "
    .
    .
    . main
    (
    )
    가
    .
    가
    ,
    가
    .

```

0.8

, (16) ‘ ,
 . ,
 .
 .
 ■ : C++ (free form) 가 .
 (space) . , (,)
 , . 가
 . 가
 #include ().
 // // , .
 /* . 가 */ ,
 . C++ 가
 . int std::ostream
 . std
 (“)
 () 가 .

<code>\n</code>	(newline)
<code>\t</code>	(tab)
<code>\b</code>	(backspace)
<code>\"</code>	('')
<code>\'</code>	('')
<code>\\</code>	\

10.2 A.2.1.3

■ *(definitions)* *(headers)* : C++

```
#include <string>
#include <iostream>
```

```
int main()
{
    return 0;
}
```

■ *main* : C++ *main* `int main` `main` 0
`return` , `main`
`return` 0 가
`main`

■ : C++

(statement)
(block)

(expression statement)

(null statement)

■ (output) : `std::cout << e` , `e` -
`std::cout` . `std::cout` ostream 가 ,

0-0. Hello, world!

0-1. 가 ?

```
3 + 4;
```

0-2.

This (") is a quote, and this (\) is a backslash.

0-3. "\t" , C++

0-4. Hello, world!

0-5. 가 ? 가 ? , 가 ?

```
#include <iostream>
int main() std::cout << "Hello, world!" << std::endl;
```

0-6. 가 ? 가 ? , 가 ?

```
#include <iostream>
int main() { { { { { std::cout << "Hello, world!" << std::endl; } } } } }
```

0-7. 가 ?

```
#include <iostream>
int main()
{
    /*
     * , /* */
     */
    std::cout << "Does this work?" << std::endl;
    return 0;
}
```


0-8. , 가 ?

```
#include <iostream>
int main()
{
    // /* */ ,
    // //
    // .
    std::cout << "Does this work?" << std::endl;
    return 0;
}
```

0-9. 가 .

0-10. (whitespace)가

Hello, world!



Chapter 1

...

0 , 가

C++ , , , , ,

, , .

,

. C++ string

, (declaration), (variable),

. 2

(control structure) .

1.1

, Hello, world!

Hello

```
//
#include <iostream>
#include <string>
int main()
{
    //
    std::cout << "Please enter your first name: ";
    //
    std::string name; // name
    std::cin >> name; // name
    //
    std::cout << "Hello, " << name << "!" << std::endl;
    return 0;
}
```

Please enter your first name:

Madimir

Hello, Madimir!

(variable)

(object)

3.2.2 , 4.2.3 , 10.6.1

가

(machine code)

```

    . ,
    . ,
    .
    , name , std::string .05 0.7
, std:: string ,
    ,
    가 , std::string 가 . <string>
    , #include 가 .

std::cout << "Please enter your first name: ";

    .
    , std::endl . std::endl
    , . ,
    .

std::string name; // name
    (definition) . name ,
std::string . , name (local
variable)가 .
가 } name (destroy), 가
    가
    .

    (interface) ,
가 (operation) . name string ( )
, string name
    .

```



```

//
#include <iostream>
#include <string>
int main()
{
    std::cout << "Please enter your first name: ";
    std::string name;
    std::cin >> name;

    //
    const std::string greeting = "Hello, " + name + "!";

    //
    const std::string spaces(greeting.size(), ' ');
    const std::string second = "* " + spaces + " *";

    //
    const std::string first(second.size(), '*');

    //
    std::cout << std::endl;
    std::cout << first << std::endl;
    std::cout << second << std::endl;
    std::cout << "* " << greeting << " *" << std::endl;
    std::cout << second << std::endl;
    std::cout << first << std::endl;
    return 0;
}
,
,
name
greeting
spaces
greeting
spaces
second
second
, first
second
*
,
#include
가
, greeting

```

```

    , 가
    , =
    가 , 10.2 string
    .
    +
    )
    가 . +가 가 . , +
    가 가 (
    가 , (overload)
    .
    const . const
    . const
    .
    가 const ,
    가 , const 가
    . , name greeting
    name , name const
    .
    (associativity) . 0 <<
    std::cout << s << t (std::cout << s) << t
    . , + (>> 가 )
    "Hello, "+name+"!" "Hello, " name "!"
    name Estragon , "Hello, "+name+"!" Hello,
    Estragon!
    , 가 가 , greeting
    .
    , 가
    .
    const std::string spaces(greeting.size(), ' ');

```



```

second          . "*" "          ,
" *"          . first
          . second          *
          . 1.1          가

```

1.3



char (built-in type)
 wchar_t “ (wide characters)”

```

string <string> . string 0
          . n , c char, is , os
          , string

```

```

std::string s;
          std::string s
std::string t = s;
          std::string t , s
          , s string
std::string z(n, c);
          std::string z , c n
          . c char , string

```

```

os << s s . , os
          (formatting) . os가

```

```

is >> s is
          . is s . , s
          . 가 . is

```

```

s + t          std::string , s          t
              . s      t          char
              .      ,          .
s.size() s

```

가

```

std::string hello = "Hello"; //
std::string stars(100, '*'); //
std::string name; //

```

}

const

■ : std::cin >> v

v

std::cin

istream

1-0.

1-1.

가 가 ? 가 ?

가 ?

```

const std::string hello = "Hello";
const std::string message = hello + ", world" + "!";

```

1-2.

가 가 ? 가 ?

가 ?

```

const std::string exclam = "!";
const std::string message = "Hello" + ", world" + exclam;

```

- 1-3. 가 ? , 가 ? ,
 가 ?

```
#include <iostream>
#include <string>
int main()
{
    { const std::string s = "a string";
      std::cout << s << std::endl; }
    { const std::string s = "another string";
      std::cout << s << std::endl; }
    return 0;
}
```

- 1-4. 가 ? }} }; 가 ?

```
#include <iostream>
#include <string>
int main()
{
    { const std::string s = "a string";
      std::cout << s << std::endl;
    { const std::string s = "another string";
      std::cout << s << std::endl; }}
    return 0;
}
```

- 1-5. 가 ? , 가 ? ,
 , .

```
#include <iostream>
#include <string>
int main()
{
    { std::string s = "a string";
    { std::string x = s + ", really";
      std::cout << s << std::endl; }
      std::cout << x << std::endl;
    }
    return 0;
}
```

1-6.

가 ?

(, Samuel Beckett).

```
#include <iostream>
#include <string>
int main()
{
    std::cout << "What is your name? ";
    std::string name;
    std::cin >> name;
    std::cout << "Hello, " << name
              << std::endl << "And what is yours? ";
    std::cin >> name;
    std::cout << "Hello, " << name
              << "; nice to meet you too!" << std::endl;
    return 0;
}
```