

CÁC ĐẶC ĐIỂM MỚI CỦA C++

C++



Microsoft®

Visual Studio®

NỘI DUNG

1

Một số lưu ý

2

Ngôn ngữ C++

3

Tham số mặc nhiên

4

Tái định nghĩa hàm

5

Quản lý bộ nhớ động

6

Truyền tham số

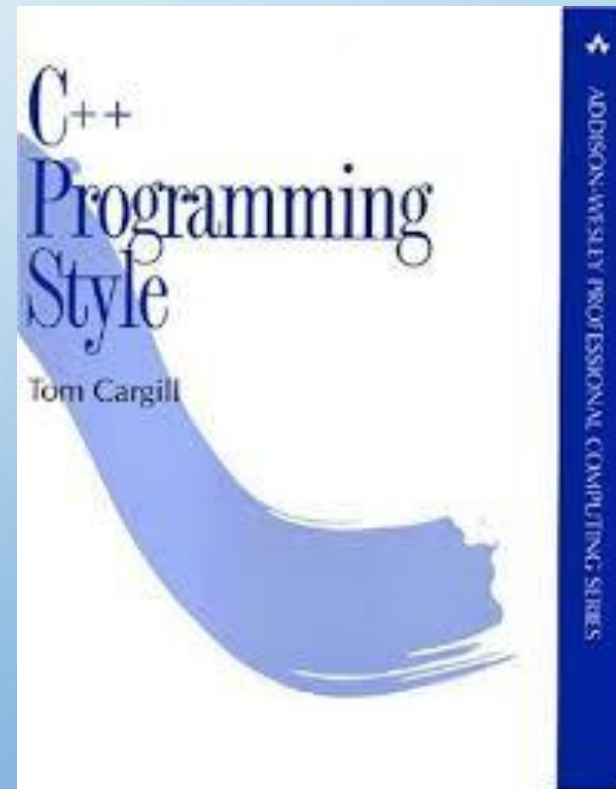
7

Inline Functions

PHONG CÁCH LẬP TRÌNH

❖ NHỮNG LƯU Ý VỀ PHONG CÁCH LẬP TRÌNH:

- ĐẶT TÊN (BIẾN, HÀM,...)
- TAB
- KHAI BÁO PROTOTYPE
- {}



BÀI TẬP C

❖ NHẬP BỐN SỐ NGUYÊN VÀ XUẤT CÁC GIÁ TRỊ VỪA NHẬP

- CÓ BAO NHIÊU CÁCH ĐỂ GIẢI QUYẾT?



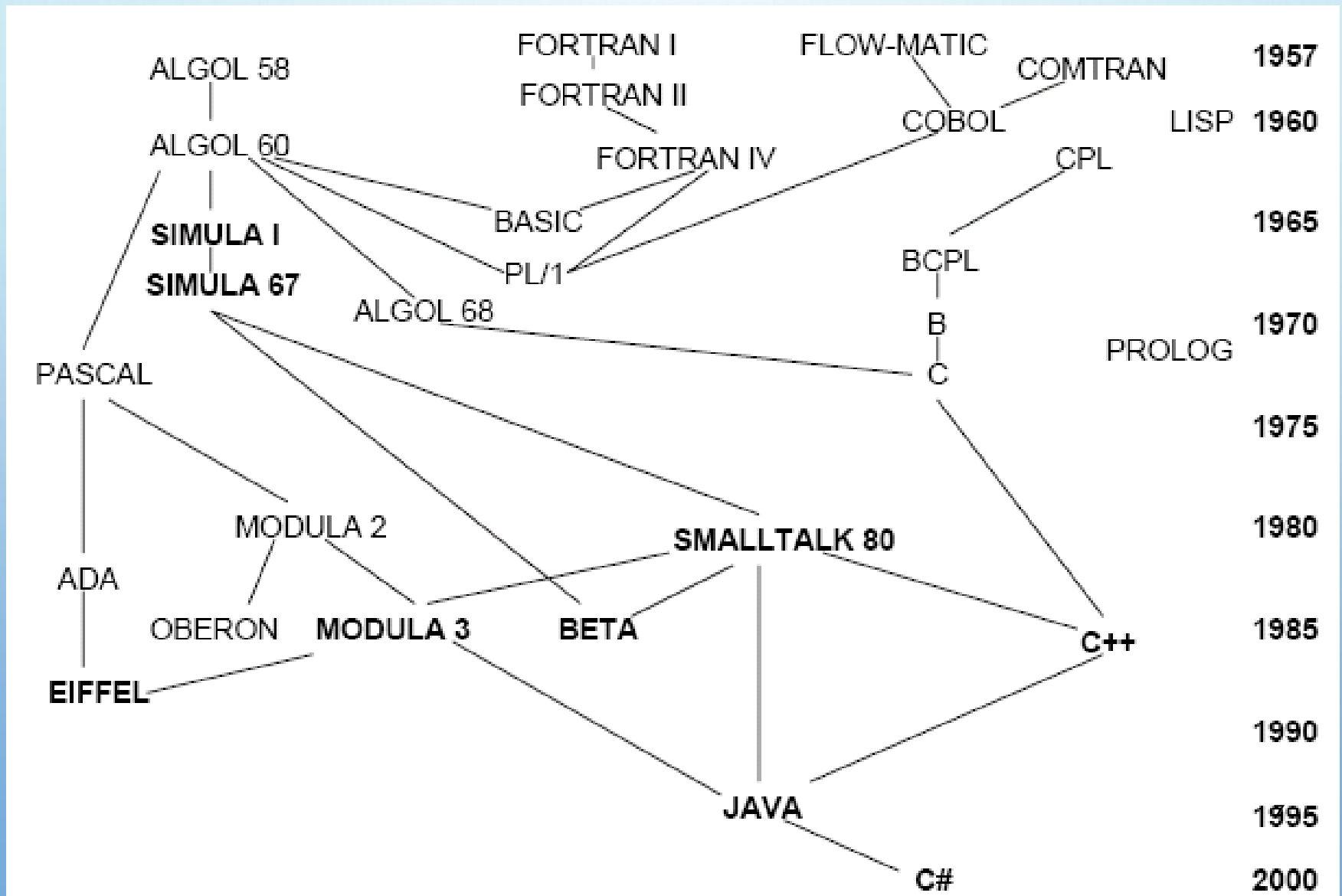
BÀI TẬP C – GIẢI

1. **DÙNG 4 BIẾN** → CÁCH DÀI NHẤT, CƠ BẢN NHẤT
2. **DÙNG MẢNG** → KHAI BÁO BIẾN GỌN HƠN, 1 LẦN THAY CHO NHIỀU LẦN
3. **DÙNG MẢNG VÀ VÒNG LẶP DO WHILE** → VIẾT CODE NHẬP GỌN HƠN, VIẾT 1 LẦN THAY CHO NHIỀU LẦN
4. **DÙNG MẢNG VÀ VÒNG LẶP FOR** → VIẾT CODE GỌN HƠN, FOR VIẾT GỌN HƠN VÒNG WHILE

BÀI TẬP C – GIẢI

5. **DÒNG MẢNG, VÒNG LẶP FOR GỘP** → VIẾT CODE GỌN HƠN, NHƯNG KHÔNG TÁCH RIÊNG ĐƯỢC 2 PHẦN NHẬP XUẤT
6. **DÒNG HÀM ĐỂ TÁCH RIÊNG PHẦN NHẬP XUẤT** → CODE CÓ THỂ TÁI SỬ DỤNG NHIỀU LẦN
7. **DÒNG FILE ĐỂ NHẬP XUẤT TỪ FILE** THAY CHO VIỆC NHẬP BẰNG BÀN PHÍM VÀ XUẤT RA MÀN HÌNH

LỊCH SỬ NGÔN NGỮ LẬP TRÌNH



LỊCH SỬ CỦA C++

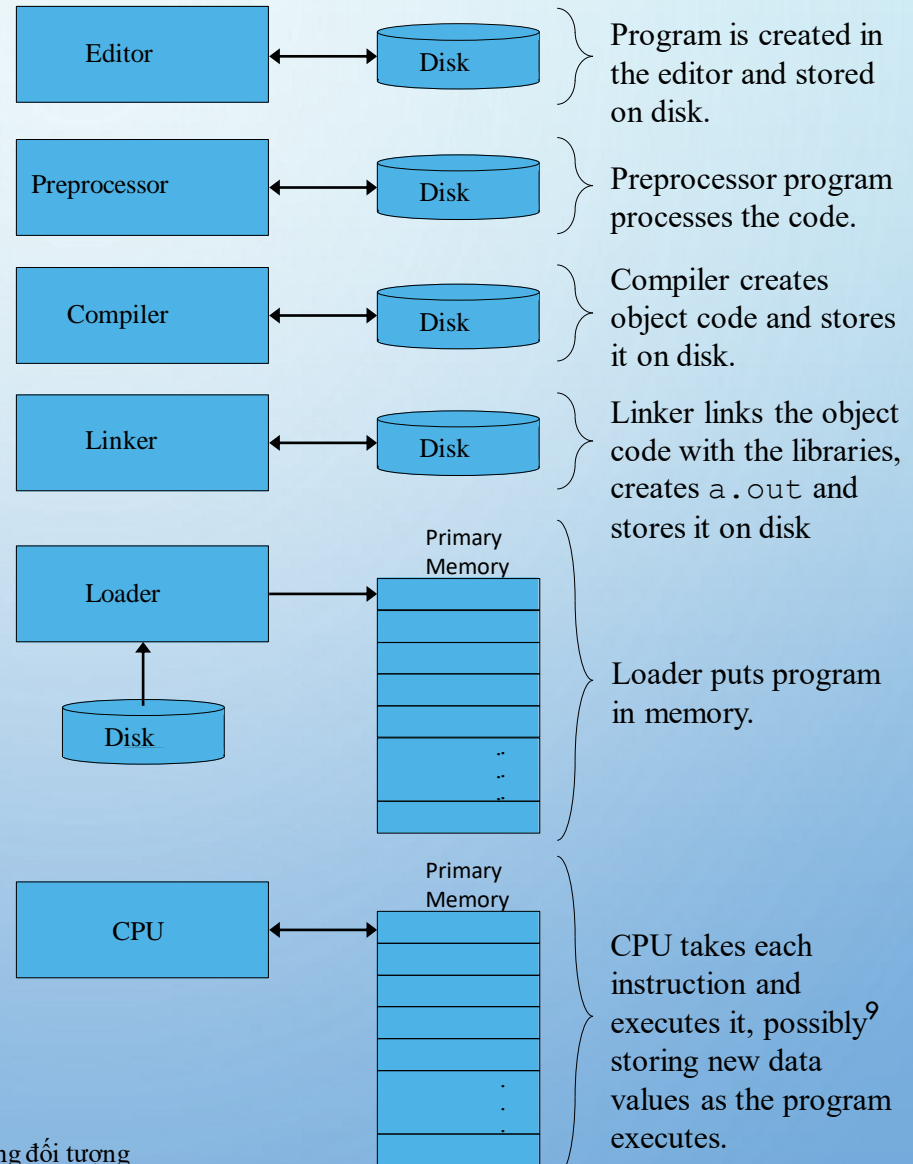
- ❖ MỞ RỘNG CỦA C
- ❖ ĐẦU THẬP NIÊN 1980: BJARNE STROUSTRUP (BELL LABORATORIES)
- ❖ CUNG CẤP KHẢ NĂNG LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG
- ❖ NGÔN NGỮ LẠI



MÔI TRƯỜNG CỦA C++

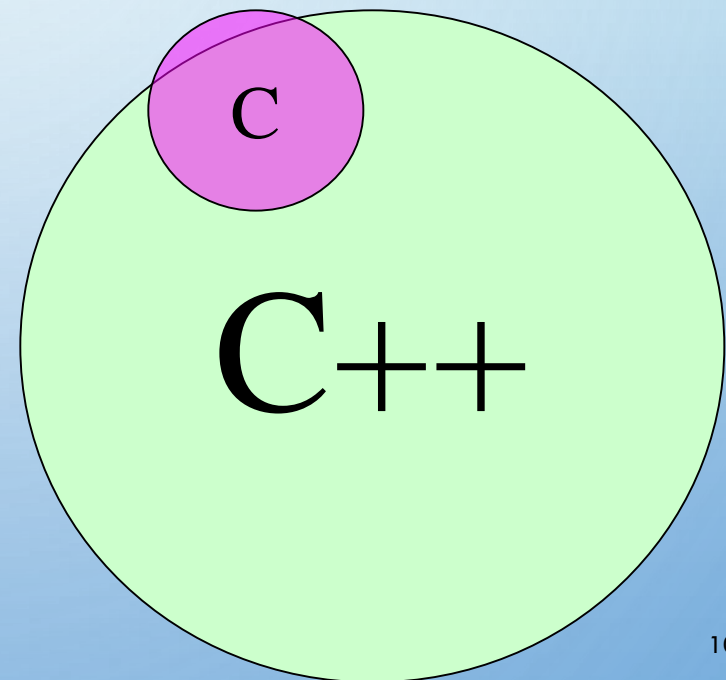
❖ BIÊN DỊCH VÀ THỰC THI CHƯƠNG TRÌNH C++:

- EDIT
- PREPROCESS
- COMPILE
- LINK
- LOAD
- EXECUTE



KHÁC BIỆT ĐỐI VỚI C

- ❖ CHÚ THÍCH
- ❖ CÁC KIỂU DỮ LIỆU
- ❖ KIỂM TRA KIỂU, ĐỔI KIỂU
- ❖ PHẠM VI VÀ KHAI BÁO
- ❖ KHÔNG GIAN TÊN
- ❖ HẰNG
- ❖ QUẢN LÝ BỘ NHỚ
- ❖ THAM CHIẾU



KHÁC BIỆT ĐỐI VỚI C

❖ PHẠM VI VÀ KHAI BÁO:

- KHÔNG GIỐNG NHƯ C, CHÚNG TA CÓ THỂ **KHAI BÁO MỘT BIẾN TẠI MỘT VỊ TRÍ BẤT KỲ** TRONG CHƯƠNG TRÌNH.
- MỘT BIẾN CHỈ CÓ TẦM TÁC DỤNG TRONG KHỐI LỆNH NÓ ĐƯỢC KHAI BÁO.
- DO ĐÓ, **C++ CUNG CẤP TOÁN TỬ ĐỊNH PHẠM VI (::)** ĐỂ XÁC ĐỊNH RÕ BIẾN NÀO ĐƯỢC SỬ DỤNG KHI XẢY RA TÌNH TRẠNG ĐỊNH NGHĨA CHỒNG MỘT TÊN BIẾN TRONG MỘT KHỐI LỆNH CON.

TOÁN TỬ PHẠM VI

❖ TOÁN TỬ PHẠM VI (::)

- THƯỜNG ĐƯỢC DÙNG ĐỂ TRUY CẬP CÁC BIẾN TOÀN CỤC TRONG TRƯỜNG HỢP CÓ BIẾN CỤC BỘ TRÙNG TÊN
- VÍ DỤ:

$Y = ::X + 3;$



TOÁN TỬ PHẠM VI

```
1 // Using the unary scope resolution operator.
2 #include <iostream>
3 #include <iomanip>
4 using namespace std;
5
6 // define global constant PI
7 const double PI = 3.14159265358979;
8 int main()
9 {
10     //define local constant PI
11     const float PI = static_cast< float >( ::PI );
```

Access the global PI with `::PI`.
Cast the global PI to a float for the local PI. This example will show the difference between float and double.

TOÁN TỬ PHẠM VI

```
12    // display values of local and global PI constants
13    cout << setprecision( 20 )
14        << " Local float value of PI = " << PI
15        << "\nGlobal double value of PI = " << ::PI<< endl;
16    return 0; // indicates successful termination
17 } // end main
```

Borland C++ command-line compiler output:

```
Local float value of PI = 3.141592741012573242
Global double value of PI = 3.141592653589790007
```

Microsoft Visual C++ compiler output:

```
Local float value of PI = 3.1415927410125732
Global double value of PI = 3.14159265358979
```


NHẬP XUẤT VỚI C++

❖CIN

❖LUỒNG NHẬP CHUẨN

❖COUT

❖LUỒNG XUẤT CHUẨN

❖CERR

❖LUỒNG THÔNG BÁO LỖI CHUẨN

NHẬP XUẤT VỚI C++

❖ CIN AND COUT (#INCLUDE <Iostream> VÀ
USING NAMESPACE std;):

```
COUT << "HEY";
```

```
CHAR NAME[10];
```

```
CIN >> NAME;
```

```
COUT<<"HEY "<<NAME<<", NICE NAME." << ENDL;
```

```
COUT << ENDL;
```

VÍ DỤ

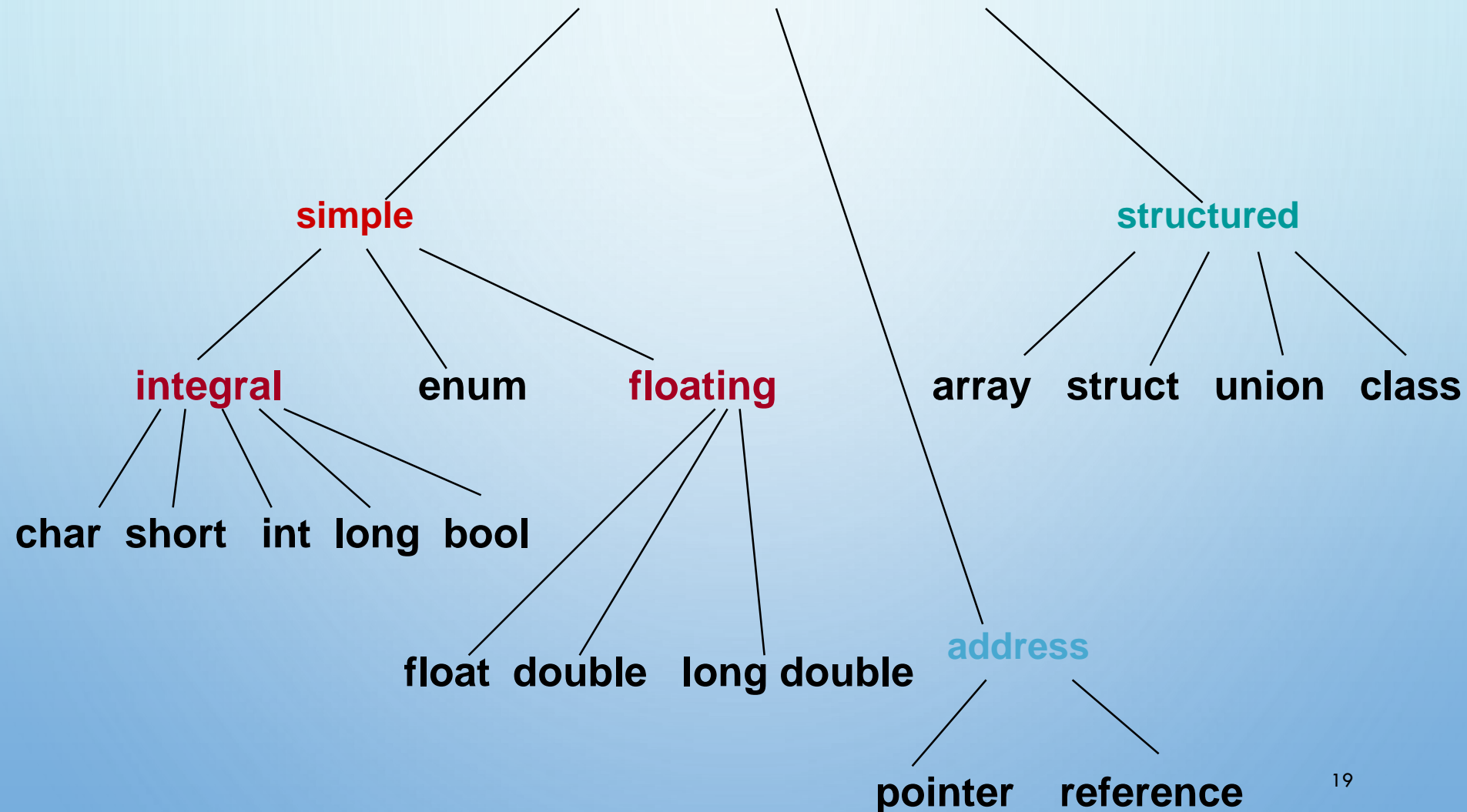
```
1 #include <iostream>
2 using namespace std;
3 // function main begins program execution
4 int main(){
5     int integer1; // first number to be input by user
6     int integer2; // second number to be input by user
7     int sum;      // variable in which sum will be stored
8     cout << "Enter first integer\n"; // prompt
9     cin >> integer1;          // read an integer
10    cout << "Enter second integer\n"; // prompt
11    cin >> integer2;          // read an integer
12    sum = integer1 + integer2; // assign result to sum
13    cout << "Sum is " << sum << endl; // print sum
12    return 0; // indicate that program ended successfully
15 } // end function main
```

VÍ DỤ

```
#INCLUDE <IOSTREAM>
USING NAMESPACE STD;
VOID MAIN() {
    INT N;
    DOUBLE D;
    CHAR S[100];

    COUT << "INPUT AN INT, A DOUBLE AND A
    STRING.";
    CIN >> N >> D >> S;
    COUT << "N = " << N << "\N";
    COUT << "D = " << D << "\N";
    COUT << "S = " << S << "\N";
}
```

CÁC KIỂU DỮ LIỆU CỦA C++



THAM SỐ MẶC NHIÊN

❖ VÍ DỤ 2:

```
void Ham1 (int a=0, int b=1) {  
    cout<<"tham so 1 = "<<a<<endl;  
    cout<<"tham so 2 = "<<b<<endl;  
}  
void main() {  
    int x=10, y=20;  
    cout << "Goi Ham1 4 lan, ta duoc : "<<endl;  
    Ham1(x,y);  
    Ham1(x);  
    Ham1(y);  
    Ham1();  
}
```


THAM SỐ MẶC NHIÊN

❖ MỤC ĐÍCH:

- GÁN CÁC GIÁ TRỊ MẶC NHIÊN CHO CÁC THAM SỐ CỦA HÀM.

❖ KHAI BÁO THAM SỐ MẶC NHIÊN:

- TẤT CẢ CÁC THAM SỐ MẶC NHIÊN PHẢI LÀ MỘT DÃY THAM SỐ LIÊN TỤC VÀ PHẢI ĐỂ Ở CUỐI HÀM.
- CHỈ CẦN ĐƯA VÀO KHAI BÁO, KHÔNG CẦN TRONG ĐỊNH NGHĨA.

❖ GỌI HÀM CÓ THAM SỐ MẶC NHIÊN:

- NẾU CUNG CẤP ĐỦ THAM SỐ → DÙNG THAM SỐ TRUYỀN VÀO.
- NẾU KHÔNG ĐỦ THAM SỐ → DÙNG THAM SỐ MẶC NHIÊN?²¹