

BÀI 03 – CON TRỎ

HAI TOÁN TỬ CƠ BẢN

1. TS. Nguyễn Tấn Trần Minh Khang
2. TS. Ngô Đức Thành
3. ThS. Võ Duy Nguyên
4. ThS. Nguyễn Hoàng Ngân

Ghi nhớ

Miền giá trị
của một biến con trỏ là
địa chỉ ô nhớ.

Toán tử &

- Yêu cầu: Hãy viết chương trình khai báo một số nguyên a. Xuất địa chỉ ô nhớ mà trình biên dịch cấp phát cho biến a này khi chương trình chạy.
- Các công việc:
 - + Khai báo một số nguyên a.
 - + Xuất địa chỉ ô nhớ mà biến a được cấp phát khi chương trình chạy.

Toán tử &

— Chương trình

```

11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a;
16.    cout<<"\n Dia chi cua bien a:";
17.    cout<<&a;
18.    return 0;
19.}

```

Toán tử &: toán tử
và (&) được sử
dụng để lấy địa chỉ
của một biến.

Toán tử &

— Xét câu lệnh

1. `int a;`

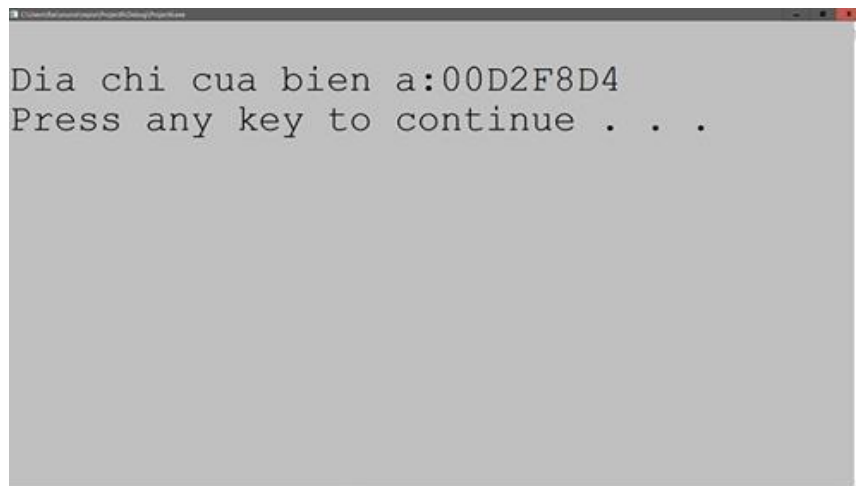
- Câu lệnh `int a;` được đọc là *a* là một biến kiểu số nguyên `int`.
- Biến *a* được khai báo như trên thì được gọi là khai báo tĩnh.
- Khi chương trình trên chạy, biến *a* sẽ được cấp phát một vùng bộ nhớ có kích thước là 4 *byte* (trên các môi trường 32 bit và 64 bit).

Toán tử &

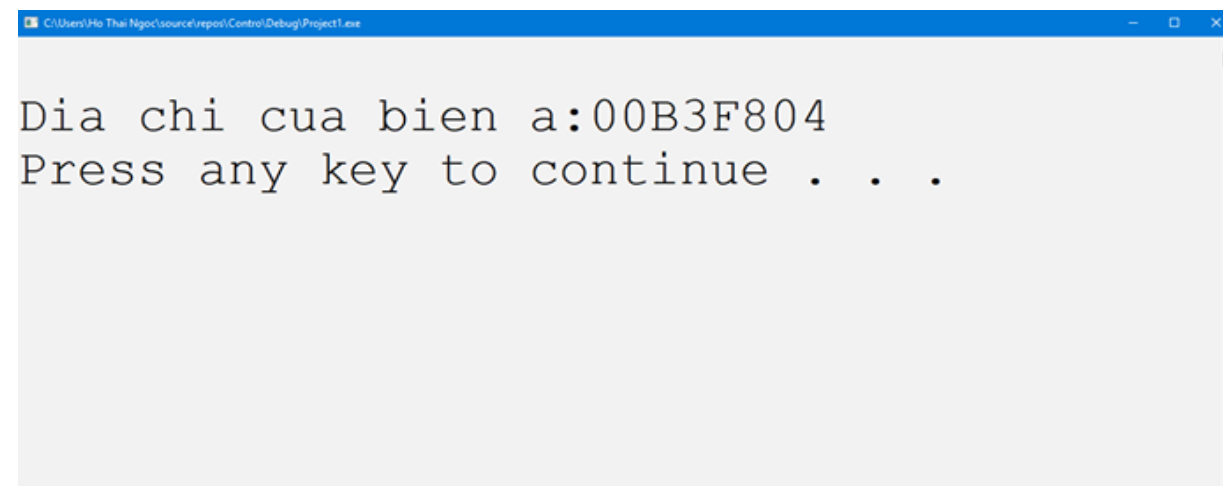
— Xét câu lệnh

1. `cout << &a;`

- Câu lệnh `cout << &a;` sẽ xuất địa chỉ ô nhớ đầu tiên của vùng nhớ có kích thước 4 byte mà compiler đã cấp phát cho biến `a`.
- Nói một cách đơn giản, câu lệnh `cout<<&a;` sẽ xuất địa chỉ của biến `a`.



```
Dia chi cua bien a:00D2F8D4
Press any key to continue . . .
```



```
Dia chi cua bien a:00B3F804
Press any key to continue . . .
```

Toán tử &

- Khi chương trình chạy mọi biến sẽ được cấp phát bộ nhớ.
- **Yêu cầu: Hãy viết và chạy chương trình sau trên máy tính.**
- **Chương trình**

```

11.#include <iostream>
12.#include <iomanip>
13.using namespace std;
14.
15.int main()
16.{
17.    int S=0;
18.    int T=1;
19.    int i=1;

```

Toán tử &

```
11. | cout << "Dia chi cua bien S:" << &S << endl;  
12. | cout << "Gia tri cua bien S:" << S << endl;  
13. |  
22. | cout << "Dia chi cua bien T:" << &T << endl;  
23. | cout << "Gia tri cua bien T:" << T << endl;  
24. |  
25. | cout << "Dia chi cua bien i:" << &i << endl;  
26. | cout << "Gia tri cua bien i:" << i << endl;  
27. | return 0;  
28. | }
```


Toán tử &

- Kết quả chạy chương trình trên trong môi trường Visual Studio 64 bit và 32 bit.

```

C:\Users\Ho Thi Ngoc\source\repos\Control\Debug\Project1.exe
Dia chi cua bien S:000000C7BF5EFAE4
Gia tri cua bien S:0
Dia chi cua bien T:000000C7BF5EFB04
Gia tri cua bien T:1
Dia chi cua bien i:000000C7BF5EFB24
Gia tri cua bien i:1
Press any key to continue . . .

```

```

C:\Users\Ho Thi Ngoc\source\repos\Control\Debug\Project1.exe
Dia chi cua bien S:00AFFD80
Gia tri cua bien S:0
Dia chi cua bien T:00AFFD74
Gia tri cua bien T:1
Dia chi cua bien i:00AFFD68
Gia tri cua bien i:1
Press any key to continue . . .

```

Toán tử &

- Yêu cầu: Hãy viết chương trình khai báo một biến con trỏ số nguyên p. Xuất địa chỉ ô nhớ mà trình biên dịch cấp phát cho nó khi chương trình chạy.

- Các công việc:
 - + Khai báo con trỏ số nguyên p.
 - + Xuất địa chỉ ô nhớ mà biến con trỏ p được cấp phát khi chương trình chạy.

Toán tử &

— Chương trình

```

1. #include <iostream>
2. using namespace std;
3. int main()
4. {
5.     int *p;
6.     cout<<"\n Dia chi cua bien p:";
7.     cout<<&p;
8.     return 0;
9. }

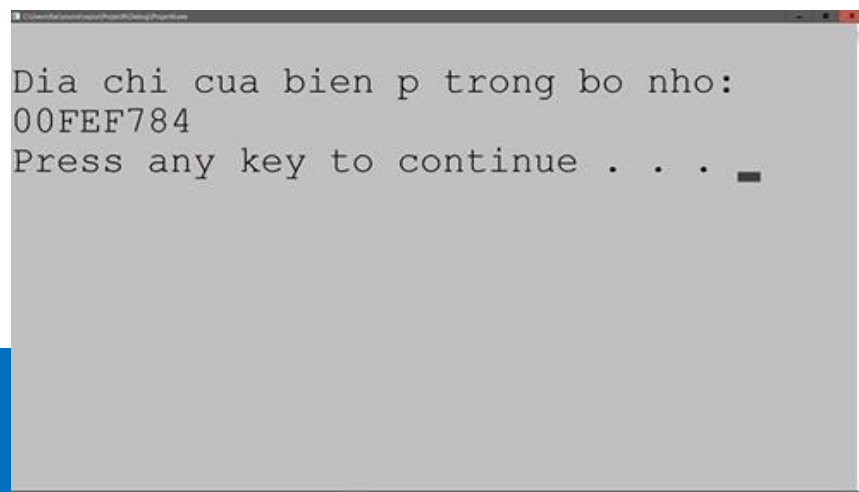
```

Toán tử &: toán tử
và (&) được sử
dụng để lấy địa chỉ
của một biến.

Toán tử &

— Phân tích:

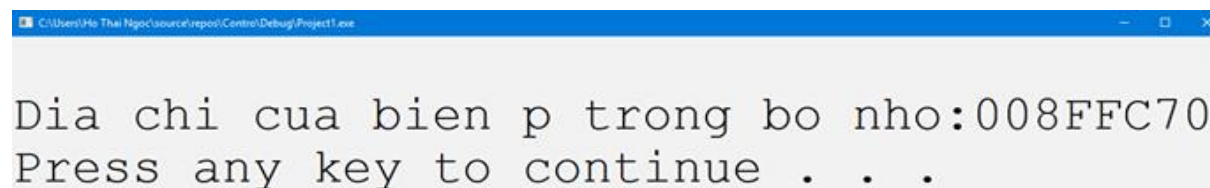
- + Câu lệnh `int *p;` được đọc và hiểu là p là biến con trỏ kiểu số nguyên, miền giá trị của biến con trỏ p là địa chỉ ô nhớ (memory address).
- + Khi chương trình trên chạy, biến con trỏ p sẽ được cấp phát một vùng bộ nhớ có kích thước là *4 byte (trong môi trường 32 bit)* hoặc *8 byte (trong môi trường 64 bit)*.
- + Kết quả chạy chương trình trên trong môi trường Visual Studio 32 bit.



```

Dia chi cua bien p trong bo nho:
00FEF784
Press any key to continue . . .

```



```

Dia chi cua bien p trong bo nho:008FFC70
Press any key to continue . . .

```

Toán tử &

- Yêu cầu: Hãy cho biết trong đoạn chương trình dưới đây kết quả của 2 dòng xuất liệu có giống nhau ko?

- Các công việc:
 - + Xác định hai dòng xuất liệu.
 - + Tư duy.

Toán tử &

— Chương trình

```

11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a;
16.    int *p;
17.    cout<<"\n Dia chi bien a:"<<&a;
18.    p = &a;
19.    cout<<"\n Dia chi bien p giu la:"<<p;
20.    return 0;
21.}

```

Cách 1: Sử dụng
con trỏ để giữ địa
chỉ của một biến.

Toán tử *

- Yêu cầu: Hãy cho biết kết quả của đoạn chương trình sau?
- Các công việc:
 - + Xác định dòng xuất liệu.
 - + Tư duy.

Toán tử *

— Chương trình

```

11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a = 10;
16.    int *p = &a;
17.    a++;
18.    cout<<"\n Gia tri tai dia chi bien ";
19.    cout<<"con          tro p dang giu la:"<<*p;
20.    return 0;
21.}

```


Toán tử *

— Chương trình

```
11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a = 10;
16.    int *p = &a;
17.    a++;
18.    cout<<"\n Gia tri tai dia chi bien ";
19.    cout<<"con          tro p dang giu la:"<<*p;
20.    return 0;
21.}
```

Toán tử *: toán tử
hoa thị (*) được sử
dụng để lấy và cập
nhật giá trị tại địa chỉ
mà biến con trỏ đang
giữ.

Toán tử *

— Chương trình

```

11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a = 10;
16.    int *p = &a;
17.    a++;
18.    cout<<"\n Gia tri tai dia chi bien ";
19.    cout<<"con          tro p dang giu la:"<<*p;
20.    return 0;
21.}

```

#1000		
#1001		
#1002		
#1003		
#1004		

Toán tử *: toán tử
hoa thị (*) được sử
dụng để lấy và cập
nhật giá trị tại địa chỉ
mà biến con trỏ đang
giữ.

Toán tử *

— Chương trình

```

11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a = 10;
16.    int *p = &a;
17.    a++;
18.    cout<<"\n Gia tri tai dia chi bien ";
19.    cout<<"con          tro p dang giu la:"<<*p;
20.    return 0;
21.}

```

#1000		
#1001		a
#1002		
#1003		
#1004		

Toán tử *: toán tử
hoa thị (*) được sử
dụng để lấy và cập
nhật giá trị tại địa chỉ
mà biến con trỏ đang
giữ.

Toán tử *

— Chương trình

```

11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a = 10;
16.    int *p = &a;
17.    a++;
18.    cout<<"\n Gia tri tai dia chi bien ";
19.    cout<<"con          tro p dang giu la:"<<*p;
20.    return 0;
21.}

```

#1000		
#1001	10	a
#1002		
#1003		
#1004		

Toán tử *: toán tử
hoa thị (*) được sử
dụng để lấy và cập
nhật giá trị tại địa chỉ
mà biến con trỏ đang
giữ.

Toán tử *

— Chương trình

```

11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a = 10;
16.    int *p = &a;
17.    a++;
18.    cout<<"\n Gia tri tai dia chi bien ";
19.    cout<<"con          tro p dang giu la:"<<*p;
20.    return 0;
21.}

```

#1000		
#1001	10	a
#1002		
#1003		p
#1004		

Toán tử *: toán tử
hoa thị (*) được sử
dụng để lấy và cập
nhật giá trị tại địa chỉ
mà biến con trỏ đang
giữ.

Toán tử *

— Chương trình

```

11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a = 10;
16.    int *p = &a;
17.    a++;
18.    cout<<"\n Gia tri tai dia chi bien ";
19.    cout<<"con          tro p dang giu la:"<<*p;
20.    return 0;
21.}

```

#1000		
#1001	10	a
#1002		
#1003	#1001	p
#1004		

Toán tử *: toán tử
hoa thị (*) được sử
dụng để lấy và cập
nhật giá trị tại địa chỉ
mà biến con trỏ đang
giữ.

Toán tử *

— Chương trình

```

11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a = 10;
16.    int *p = &a;
17.    a++;
18.    cout<<"\n Giá trị tại địa chỉ biến";
19.    cout<<"con trỏ p đang giữ là:"<<*p;
20.    return 0;
21.}

```

#1000		
#1001	10 11	a
#1002		
#1003	#1001	p
#1004		

Toán tử *: toán tử
hoa thị (*) được sử
dụng để lấy và cập
nhật giá trị tại địa chỉ
mà biến con trỏ đang
giữ.

Toán tử *

- Yêu cầu: Hãy cho biết kết quả của đoạn chương trình sau?
- Các công việc:
 - + Xác định các dòng xuất liệu.
 - + Tư duy.

Toán tử *

```

11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a = 10;
16.    int *p = &a;
17.    cout<<"\n Giá trị tại địa chỉ con trỏ p đang giữ:"<<*p;
18.    *p = 15;
19.    cout<<"\n Bien a:"<<a;
20.    cout<<"\n Giá trị tại địa chỉ con trỏ p đang giữ:"<<*p;
21.    return 0;
22.}

```

Toán tử *

```
11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a = 10;
16.    int *p = &a;
17.    cout<<"\n Giá trị tại địa chỉ con trỏ p đang giữ:"<<*p;
18.    *p = 15;
19.    cout<<"\n Bien a:"<<a;
20.    cout<<"\n Giá trị tại địa chỉ con trỏ p đang giữ:"<<*p;
21.    return 0;
22.}
```

Toán tử *: toán tử
hoa thị (*) được sử
dụng để lấy và cập
nhật giá trị tại địa chỉ
mà biến con trỏ đang
giữ.

Toán tử *

```

11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a = 10;
16.    int *p = &a;
17.    cout<<"\n Giá trị tại địa chỉ con trỏ p đang giữ:"<<*p;
18.    *p = 15;
19.    cout<<"\n Bien a:"<<a;
20.    cout<<"\n Giá trị tại địa chỉ con trỏ p đang giữ:"<<*p;
21.    return 0;
22.}

```

#1000		
#1001		
#1002		
#1003		
#1004		

Toán tử *: toán tử
hoa thị (*) được sử
dụng để lấy và cập
nhật giá trị tại địa chỉ
mà biến con trỏ đang
giữ.

Toán tử *

```

11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a = 10;
16.    int *p = &a;
17.    cout<<"\n Giá trị tại địa chỉ con trỏ p đang giữ:"<<*p;
18.    *p = 15;
19.    cout<<"\n Bien a:"<<a;
20.    cout<<"\n Giá trị tại địa chỉ con trỏ p đang giữ:"<<*p;
21.    return 0;
22.}

```

#1000		a
#1001		
#1002		
#1003		
#1004		

Toán tử *: toán tử
hoa thị (*) được sử
dụng để lấy và cập
nhật giá trị tại địa chỉ
mà biến con trỏ đang
giữ.

Toán tử *

```

11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a = 10;
16.    int *p = &a;
17.    cout<<"\n Giá trị tại địa chỉ con trỏ p đang giữ:"<<*p;
18.    *p = 15;
19.    cout<<"\n Bien a:"<<a;
20.    cout<<"\n Giá trị tại địa chỉ con trỏ p đang giữ:"<<*p;
21.    return 0;
22.}

```

#1000	10	a
#1001		
#1002		
#1003		
#1004		

Toán tử *: toán tử
hoa thị (*) được sử
dụng để lấy và cập
nhật giá trị tại địa chỉ
mà biến con trỏ đang
giữ.

Toán tử *

```

11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a = 10;
16.    int *p = &a;
17.    cout<<"\n Giá trị tại địa chỉ con trỏ p đang giữ:"<<*p;
18.    *p = 15;
19.    cout<<"\n Bien a:"<<a;
20.    cout<<"\n Giá trị tại địa chỉ con trỏ p đang giữ:"<<*p;
21.    return 0;
22.}

```

#1000	10	a
#1001		
#1002		
#1003		p
#1004		

Toán tử *: toán tử
hoa thị (*) được sử
dụng để lấy và cập
nhật giá trị tại địa chỉ
mà biến con trỏ đang
giữ.

Toán tử *

```

11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a = 10;
16.    int *p = &a;
17.    cout<<"\n Giá trị tại địa chỉ con trỏ p đang giữ:"<<*p;
18.    *p = 15;
19.    cout<<"\n Bien a:"<<a;
20.    cout<<"\n Giá trị tại địa chỉ con trỏ p đang giữ:"<<*p;
21.    return 0;
22.}

```

#1000	10	a
#1001		
#1002		
#1003	#1000	p
#1004		

Toán tử *: toán tử
hoa thị (*) được sử
dụng để lấy và cập
nhật giá trị tại địa chỉ
mà biến con trỏ đang
giữ.

Toán tử *

```

11.#include <iostream>
12.using namespace std;
13.int main()
14.{
15.    int a = 10;
16.    int *p = &a;
17.    cout<<"\n Giá trị tại địa chỉ con trỏ p đang giữ:"<<*p;
18.    *p = 15;
19.    cout<<"\n Bien a:"<<a;
20.    cout<<"\n Giá trị tại địa chỉ con trỏ p đang giữ:"<<*p;
21.    return 0;
22.}

```

#1000	10 15	a
#1001		
#1002		
#1003	#1000	p
#1004		

Toán tử *: toán tử
hoa thị (*) được sử
dụng để lấy và cập
nhật giá trị tại địa chỉ
mà biến con trỏ đang
giữ.

Chúc các bạn học tốt