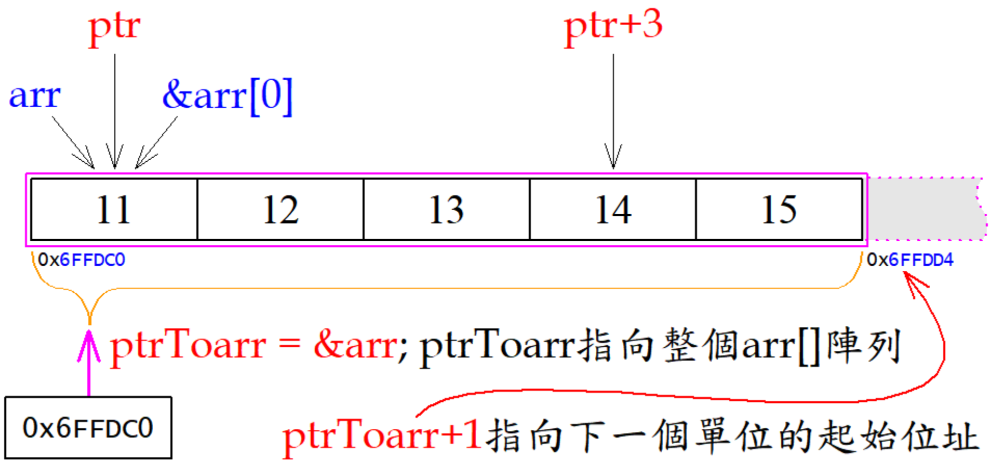
指向整個陣列的指標 (1/3)

- 相較於指向陣列個別元素的指標，&arrayName用於指向整個陣列
  - 對應的資料型態例
    ```
    int  (*ptr1)[5];
    char (*ptr2)[15];
    ```

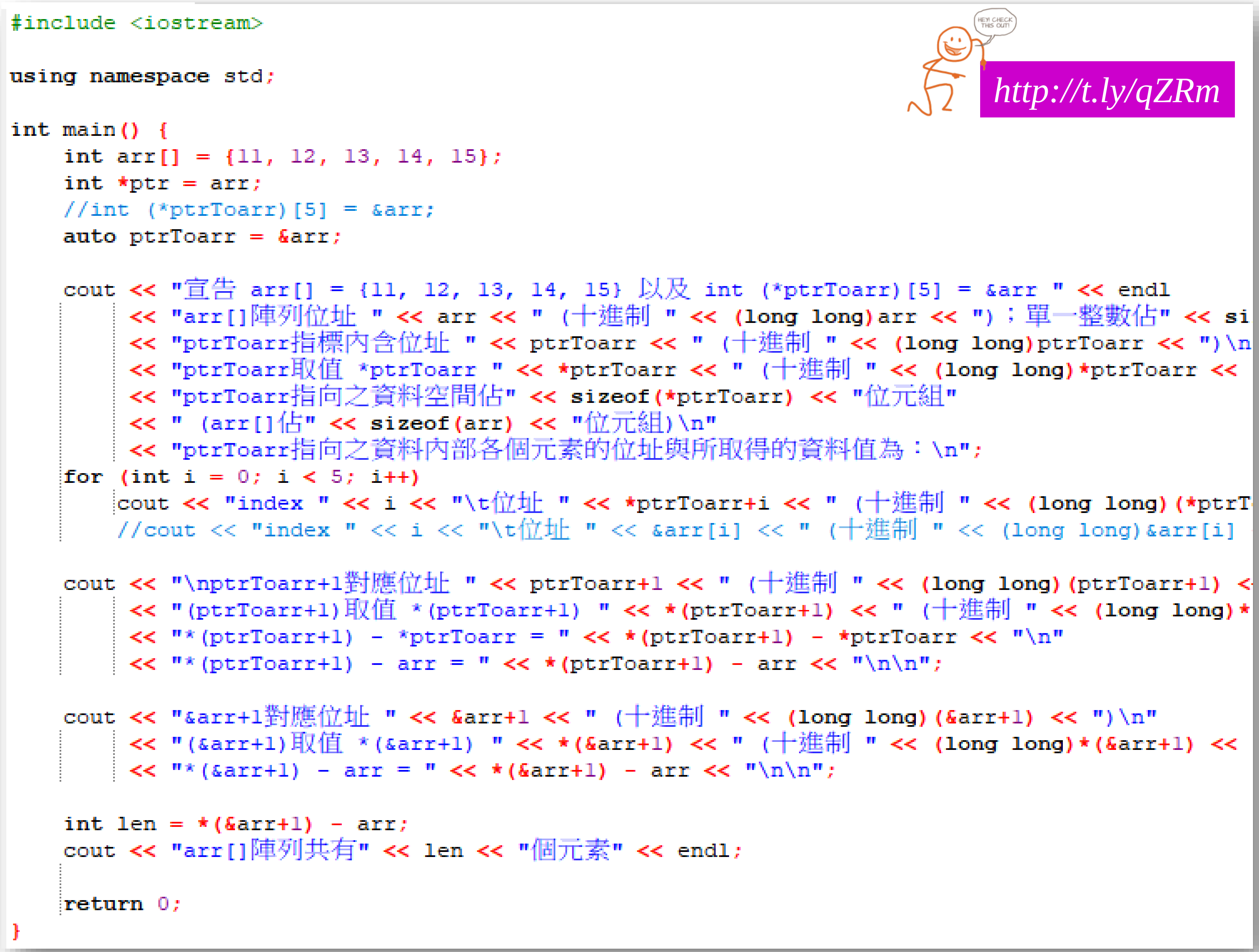
//若寫為 int *ptr1[5] 意為宣告5個指向整數空間
//的指標， 以陣列形式配置

- 例
  - int arr[] = {11,12,13,14,15};
    int *ptr = arr;
    int (*ptrToarr)[5] = &arr;



ptr
ptr+3
arr
&arr[0]
11
12
13
14
15
0x6FFDC0
0x6FFDD4
ptrToarr = &arr; ptrToarr指向整個arr[]陣列
0x6FFDC0
ptrToarr+1指向下一個單位的起始位址

  - &arr[0]代表arr[]陣列第0元素的位址
  - ptrToarr指向整個arr[]陣列
  - ptrToarr+1代表以arr[]所佔空間為一單位，取得下一個單位的起始位址
  - ptrToarr猶如指標的指標，透過二次取值運算才讀取到儲存的資料



http://t.ly/qZRm

```
#include <iostream>

using namespace std;

int main() {
    int arr[] = {11, 12, 13, 14, 15};
    int *ptr = arr;
    //int (*ptrToarr)[5] = &arr;
    auto ptrToarr = &arr;

    cout << "宣告 arr[] = {11, 12, 13, 14, 15} 以及 int (*ptrToarr)[5] = &arr " << endl
         << "arr[]陣列位址 " << arr << " (十進制 " << (long long)arr << ")；單一整數佔" << si
         << "ptrToarr指標內含位址 " << ptrToarr << " (十進制 " << (long long)ptrToarr << ")\n
         << "ptrToarr取值 *ptrToarr " << *ptrToarr << " (十進制 " << (long long)*ptrToarr <<
         << "ptrToarr指向之資料空間佔" << sizeof(*ptrToarr) << "位元組"
         << " (arr[]佔" << sizeof(arr) << "位元組)\n"
         << "ptrToarr指向之資料內部各個元素的位址與所取得的資料值為：\n";
    for (int i = 0; i < 5; i++)
        cout << "index " << i << "\t位址 " << *ptrToarr+i << " (十進制 " << (long long)(*ptrT
        //cout << "index " << i << "\t位址 " << &arr[i] << " (十進制 " << (long long)&arr[i]

    cout << "\nptrToarr+1對應位址 " << ptrToarr+1 << " (十進制 " << (long long)(ptrToarr+1) <
         << "(ptrToarr+1)取值 *(ptrToarr+1) " << *(ptrToarr+1) << " (十進制 " << (long long)*
         << "*(ptrToarr+1) - *ptrToarr = " << *(ptrToarr+1) - *ptrToarr << "\n"
         << "*(ptrToarr+1) - arr = " << *(ptrToarr+1) - arr << "\n\n";

    cout << "&arr+1對應位址 " << &arr+1 << " (十進制 " << (long long)(&arr+1) << ")\n"
         << "(&arr+1)取值 *(&arr+1) " << *(&arr+1) << " (十進制 " << (long long)*(&arr+1) <<
         << "*(&arr+1) - arr = " << *(&arr+1) - arr << "\n\n";

    int len = *(&arr+1) - arr;
    cout << "arr[]陣列共有" << len << "個元素" << endl;

    return 0;
}
```

指向整個陣列的指標 (3/3)

- ***(&arrayName+1) – arrayName** 可用於判讀陣列元素個數，但如果是二維陣列，同樣的原理也適用嗎？

```
#include <iostream>

using namespace std;

int main() {
      int arr[3][5] = { {11, 12, 13, 14, 15},
                        {21, 22, 23, 24, 25},
                        {31, 32, 33, 34, 35}  };

      int len = *(&arr+1) - arr;
      cout << "arr[]陣列共有" << len << "個元素" << endl;

      return 0;
}
```