C++程式設計

字串

C型態字串

C型態字串(C-style string): 即**字元陣列**， 是早期C語言處理字串的方式。

如：

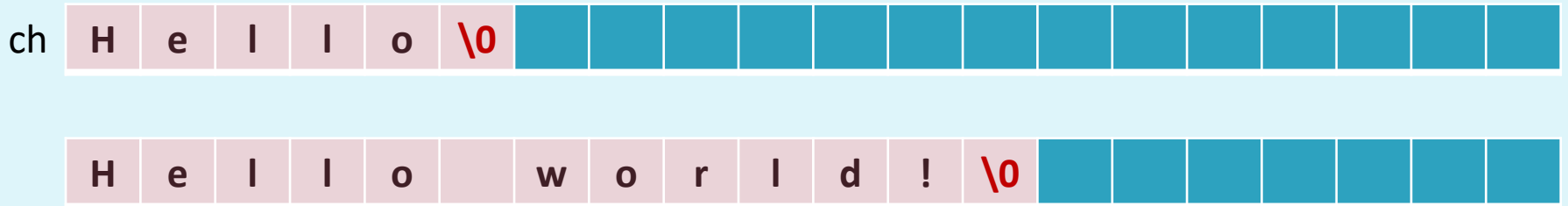
```
char ch[20];
```

即宣告1個包含20個元素的字元陣列

C型態字串

字元陣列可用以下2種方式設定初值

```
char ch[20]={'H','e','l','l','o','\0'};
char ch[20]="Hello";
```



- 字元常數以單引號包圍(‘)，字串常數以雙引號包圍(“)
- 字串常數儲存在記憶體中，以**字串結束字元‘\0’**結尾

C型態字串

```
1   #include <iostream>
2   using namespace std;
3
4   int main()
5   {
6       char a[]="My friend";
7       char b='c',str[]="c";
8       cout << "sizeof(a)=" << sizeof(a) << endl;
9       cout << "sizeof(b)=" << sizeof(b) << endl;
10      cout << "sizeof(str)=" << sizeof(str) << endl;
11  }
```

```
sizeof(a)=10
sizeof(b)=1
sizeof(str)=2
```

C型態字串 輸入與輸出

```
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5     char name[10];
6     cout << "What is your name? ";
7     cin >> name;
8     cout << "Hello, " << name << endl;
9  }
```

```
What is your name? Jack
Hello, Jack
```

```
What is your name? Michael Jordan
Hello, Michael
```

cin讀到空白即結束讀取

C型態字串 輸入與輸出

若輸入帶有空白的字串，可使用cin.getline()函數

```
cin.getline(字串名稱, 最大字串長度, 字串結束字元)
```

- 字串結束字元預設為換行字元'\n'，若不更改可省略
- cin.getline(str, 10); //輸入一個長度為10的字串至str中

C型態字串 輸入與輸出

```
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5     char name[15];
6     cout << "What is your name? ";
7     cin.getline(name,15);
8     cout << "Hello, " << name << endl;
9  }
```

```
What is your name? Michael Jordan
Hello, Michael Jordan
```

讀入帶有空白的字串

C型態字串 輸入與輸出

```
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5     int age;
6     char name[15];
7     cout << "How old are you? ";
8     cin >> age;
9     cout << "What is your name? ";
10    cin.getline(name,15);
11    cout << name << " is " << age << " years old!" << endl;
12 }
```

```
How old are you? 15
What is your name?  is 15 years old!
```

cin讀入age時,將\n留在序列中,cin.getline()讀入字串時接收\n,成為空白字串

C型態字串 輸入與輸出

```
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5     int age;
6     char name[15];
7     cout << "How old are you? ";
8     cin >> age;
9     cin.get();
10    cout << "What is your name? ";
11    cin.getline(name,15);
12    cout << name << " is " << age << " years old!" << endl;
13 }
```

```
How old are you? 15
What is your name? Charlie Brown
Charlie Brown is 15 years old!
```

使用cin.get()函數將前面輸入留下的\n吸收

C++型態字串

C++型態字串(C++-style string): C++提供另一種使用字串的方式， 使用**string類別**。

需使用string含括檔：

```
#include <string>
```

宣告及設值格式：

string 字串名稱;
字串名稱 = “字串常數”;

```
string str1;
str1 = "Hello C++!";
```

宣告並設值：

string 字串名稱 = “字串常數”;

```
string str2 = "Hello C++!";
```

C++型態字串 字串的運算

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  int main()
6  {
7      string first="Charlie";
8      string last="Brown";
9      cout << "full name: " << first+" "+last << endl;
10     first+=" ";
11     first+=last;
12     cout << "full name: " << first << endl;
13 }
```

```
full name: Charlie Brown
full name: Charlie Brown
```

C++型態字串 輸出與輸入

若輸入帶有空白的字串，可使用getline()函數

```
getline(cin, 字串物件)
```

- getline(cin, str); //輸入一個字串至str中

C++型態字串 輸出與輸入

```cpp
#include <iostream>
#include <string>
using namespace std;

int main()
{
    int num;
    string proverb;
    cout << "次數: ";
    cin >> num;
    cin.get();
    cout << "文字: ";
    getline(cin,proverb);
    for(int i=1;i<=num;i++)
     cout << proverb << endl;
}
```

```
次數: 3
文字: Practice makes perfect
Practice makes perfect
Practice makes perfect
Practice makes perfect
```

C++型態字串 字串函數

函數	說明
str1.at(i)	從str取出第i個字元(i起始值為0)
str1.append(str2)	將str2附加在str1之後
str1.append(str2, i, l)	從str2第i個字元開始,取出l個字元附加在str1之後
str1.substr(i)	取出str1中第i個到最後一個字元
str1.substr(i, l)	從str1的第i個開始,取出l個字元
str1.length()	求str1的長度

更多字串函數
http://www.cplusplus.com/reference/string/string/

C++型態字串 字串函數

```cpp
#include <iostream>
#include <string>
using namespace std;

int main()
{
    string str1="Charlie ";
    string str2="Brown";
    string str3=", 2015/12/25";
    cout << "str1.append(str2)" << endl;
    str1.append(str2);
    cout << str1 << endl;
    cout << "str1.append(str3,0,6)" << endl;
    str1.append(str3,0,6);
    cout << str1 << endl;
    cout << "str1.substr(8)" << endl;
    cout << str1.substr(8) << endl;
    cout << "str1.length()" << endl;
    cout << str1.length() << endl;
}
```

```
str1.append(str2)
Charlie Brown
str1.append(str3,0,6)
Charlie Brown, 2015
str1.substr(8)
Brown, 2015
str1.length()
19
```

C++型態字串 字串函數

```
1  #include<iostream>
2  using namespace std;
3  int main()
4  {
5      string s;
6      cin >> s;
7      for(int i=s.length()-1;i>=0;i--)
8          cout << s.at(i);
9      cout << endl;
10
11 }
```