國立屏東大學 資訊工程學系 物件導向程式設計

11. 陣列

基本上C++的陣列就如同C語言中的陣列一樣，並沒有特別不同之處，除了C++的陣列可以支援較多的資料型態以及物件(當然這都是來自於C++的新功能)。關於陣列的基礎，可參考C語言的陣列，在此不予贅述。

11.1 動態陣列

我們可以透過new與delete來動態地建立與刪除(或者說回收)陣列。請參考下面的例子：

```
int *p = new int [10]; //動態配置產生一塊存放10個整數的陣列空間，並讓p指向它。

p[0]=3;
p[1]=2;

delete [] p;             //不再使用時將其刪除
```

如果是二維的陣列，不可以用int * * p = new int[3][2];這種方式宣告，你必須以int * * p = new int * [3];先宣告一個pointer-to-pointer並配置三個指向陣列的指標給它。然後再配合for迴圈，為每個陣列動態地配置空間，請參考以下的例子：

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

#define ROW 3
#define COL 2

int main()
{
  int **array2d = new int * [ROW];
  for(int i=0;i<ROW;i++)
  {
    array2d[i] = new int [COL];
  }

  for(int i=0;i<ROW;i++)
    for(int j=0;j<COL;j++)
      array2d[i][j] = (i+1)*(j+1);

  for(int i=0;i<ROW;i++)
  {
```

```
    for(int j=0;j<COL;j++)
    {
      cout << array2d[i][j] << " " ;
    }
    cout << endl;
  }

  for(int i=0;i<ROW; i++)
    delete [] array2d[i];

  delete [] array2d;

  return 0;
}
```

特別注意其記憶體空間的回收，也不支援delete [][]的方法，你必須手動逐一的回收。

From:
https://junwu.nptu.edu.tw/dokuwiki/ - **Jun Wu的教學網頁**
國立屏東大學資訊工程學系
CSIE, NPTU
Total: 334337

Permanent link:
https://junwu.nptu.edu.tw/dokuwiki/doku.php?id=cpp:arrays

Last update: **2020/03/19 14:56**