

JT-70 依不同規則的排序結果

題目

輸入十個整數，依以下規則分別印出。1) 原來順序，2) 從小到大，3) 從大到小，4) 偶數在前，從小到大；奇數在後，從大到小。

解題思惟

1. 這一題是課堂上講解函數指標的時候用的例子。基本上寫一個氣泡排序法，但是檢查大小的部份，使用自訂的比較函數處理。函數宣告如下：

```
void general_bubble(int *arr, int len, int (*comp)(int, int));
```

最後一個參數 comp 是函數指標，指向兩個整數輸入，一個整數輸出的函數。

2. 撰寫比較函數的時候，要記得傳入的兩個參數，如果順序正確的話，要回傳 0（不用交換）；如果順序不正確的話，要回傳 1（或任何非 0 的數）。
3. 比較順序的時候，例如 $x > y$ 要回傳 1，否則回傳 0，雖然可以寫成

```
if (x>y) return 1;
    else return 0;
```

但有更簡便的寫法，直接寫成

```
return (x>y);
```

就可以了。（實際上 $x > y$ 時，可能是回傳 -1，因為 C 習慣使用 -1 來表達 True，但這也是非 0 的數，意義相同）

程式碼

```
#include <stdio.h>
#include <stdlib.h>

int asc(int x, int y); // From small to large
int desc(int x, int y); // From large to small
int myorder(int x, int y); // Even first, small to large, Odd later, large to small
void general_bubble(int *arr, int len, int (*comp)(int, int));
void printarr(int *arr, int len);

int main()
{
    int i, a[10];
    for (i=0; i<10; i++) scanf("%d", a+i);
    printf("Original: "); printarr(a, 10);
    general_bubble(a, 10, asc);
    printf("ASC order: "); printarr(a, 10);
    general_bubble(a, 10, desc);
    printf("DESC order: "); printarr(a, 10);
    general_bubble(a, 10, myorder);
    printf("My special order: "); printarr(a, 10);
    return 0;
}
```

```

void general_bubble(int *arr, int len, int (*comp)(int, int))
{
    int i, j, temp;
    for (i = 0; i < len - 1; i++)
        for (j = 0; j < len - 1 - i; j++)
            if ((*comp)(arr[j], arr[j + 1]))    // If order wrong, swap
            {
                temp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = temp;
            }
}

int asc(int x, int y) // From small to large
{
    return x>y;    // If order wrong, then return true (non-zero)
}

int desc(int x, int y)    // From large to small
{
    return x<y;    // If order wrong, then return true (non-zero)
}

int myorder(int x, int y) // Even first, small to large, Odd later, large to small
{
    if (x%2==0 && y%2==0) return x>y;    // Both are x, y are even
    if (x%2==0 && y%2==1) return 0;    // x even, y odd
    if (x%2==1 && y%2==0) return 1;    // x odd, y even
    return x<y;
    // x, y are both odd
}

void printarr(int *arr, int len)
{
    int i;
    for (i=0; i<len; i++) printf(" %d", arr[i]);
    printf("\n");
}

```