程式技巧

函式 :~ 程式習慣 $

可以幫助更快的 debug 、保證正確率、增加可讀性

* 縮排
* 架構
* 註解

函式 :~ 程式習慣 / 縮排 $

```cpp
// good version
for (int i=0 ; i<n ; i++){
    for (int j=0 ; j<n ; j++){
        if (i==j){
            cout << "@";
        }else{
            cout << "#";
        }
    }
    cout << endl;
}
```

```cpp
// bad version
for (int i=0 ; i<n ; i++){
for (int j=0 ; j<n ; j++){
if (i==j){
cout << "@";
}else{
cout << "#";
}
}
cout << endl;
}
```

函式 :~ 程式習慣 / 架構 $

一般來說，寫一份題目的架構如下

- declare　　　（宣告）
- function　　 （函式）
- init　　　　（初始化）
- input　　　 （輸入）
- process　　 （處理）
- output　　　（輸出）

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

// declare

// function

int main(){
    // init

    // input

    // process

    // output

    return 0;
}
```

函式 :~ 程式習慣 / 註解 $

```cpp
// declare
int n, k;
int a[105]; // 儲存輸入
int now=0;  // 維護當前的的怒氣值總和
int ans=0;  // 維護總節省時間
```

```cpp
// function
bool check(int mid){ // 回傳這個數值是否合法
    return mid>=k;
}
```

函式 :~debug$

- 重新整理一次程式碼

- 印出所有重要變數

- 設定中斷點

函式 :~debug/ 印變數 $

```cpp
// input
cin >> x >> y >> r;
tmp_x=x*10000;
tmp_y=y*10000;
tmp_r=r*10000;

ll=ceil((tmp_x-tmp_r)/10000.0)*10000;
rr=floor((tmp_x+tmp_r)/10000.0)*10000;

// preview
cout << "x: " << tmp_x << " y: " << tmp_y << " r: " << tmp_r << endl;
cout << ll << " " << rr << endl;
cout << "\n===============\n\n";
```

函式 :~debug/ 印變數 $

```cpp
// get answer
for (int i=ll ; i<=rr ; i+=10000){
    // sensor line
    ans+=get_value(i);
    cout << "i: " << i << " get: " << get_value(i);
}
```

函式 :~debug/ 中斷點 $

```cpp
cin >> n;
for (int i = 0; i < n; i = i++) {
    cin >> a >> b >> c >> d;
    arr[t] = a;
    arr[t + 1] = b;
    arr[t + 2] = c;
    arr[t + 3] = d;
    t = t + 4;
}
int x = 0;
for (int j = 0; j < n; j++) {
    if ((arr[x] + arr[x + 1] + arr[x + 2]
        A = "Yes";
    }
    else if (arr[x] > 6 || arr[x + 1] > 6
        A = "Yes";
    }
    else if (arr[x] < 3 || arr[x + 1] < 3
        A = "Yes";
    }
```

```cpp
cin >> n;
for (int i = 0; i < n; i = i++) {
    cin >> a >> b >> c >> d;
    arr[t] = a;
    arr[t + 1] = b;
    arr[t + 2] = c;
    arr[t + 3] = d;
    t = t + 4;
}
// int x = 0;
// for (int j = 0; j < n; j++) {
//     if ((arr[x] + arr[x + 1] + arr[x +
//         A = "Yes";
//     }
//     else if (arr[x] > 6 || arr[x + 1] >
//         A = "Yes";
//     }
//     else if (arr[x] < 3 || arr[x + 1] <
//         A = "Yes";
//     }
```

函式 :~debug/ 中斷點 $

```cpp
cin >> n;
for (int i = 0; i < n; i = i++) {
    cin >> a >> b >> c >> d;
    // arr[t] = a;
    // arr[t + 1] = b;
    // arr[t + 2] = c;
    // arr[t + 3] = d;
    // t = t + 4;
}
// int x = 0;
// for (int j = 0; j < n; j++) {
//     if ((arr[x] + arr[x + 1] + arr[x +
//         A = "Yes";
//     }
//     else if (arr[x] > 6 || arr[x + 1] >
//         A = "Yes";
//     }
//     else if (arr[x] < 3 || arr[x + 1] <
//         A = "Yes";
//     }
```