

Contents

1 Shell Script

2 Libraries

2.1	cstdlib	1
2.2	algorithm	1
2.3	map	1
2.4	set	1
2.5	vector	1
2.6	string	1

3 Algorithms

3.1	最短路	1
3.1.1	Bellman-Ford	1
3.1.2	Dijkstra's	1
3.2	LIS - Longest Increasing Subsequence	1

4 Formula

4.1	thm	1
-----	-----	---

1 Shell Script

```
1 #!/bin/bash
2 clear
3 g++ $1.cpp -DDBG -o $1
4 if [[ "$?" == "0" ]]; then
5     echo Running
6     ./$1 <$1.in> $1.out
7     echo END
8 fi
```

2 Libraries

2.1 cstdlib

```
1 #include <cstdlib>
2 using namespace std;
3 {
4     // Function: String conversion
5     double atof(const char* str);
6         // char* 轉 double
7     int atoi(const char * str);
8         // char* 轉 int
9     long int atol(const char * str);
10        // char* 轉 long int
11    long long int atoll(const char * str);
12        // char* 轉 long long int
13    double strtod(const char* str, char** endptr);
14        // char* 轉 double;
15    float strtof(const char* str, char** endptr);
16        // char* 轉 float
17    long int strtol(const char* str, char** endptr,
18        int base);
19        // char*(base) 轉 long int
20        // 且指向轉換子字串之末
21    long double strtold(const char* str, char**
22        endptr);
23        // char*(base) 轉 long double
24        // 且指向轉換子字串之末
25    long long int strtoll(const char* str, char**
26        endptr, int base);
27        // char*(base) 轉 long long int
28        // 且指向轉換子字串之末
29    unsigned long int strtoul(const char* str, char**
30        endptr, int base);
31        // char*(base) 轉 unsigned long int
32        // 且指向轉換子字串之末
33    unsigned long long int strtoull(const char* str,
34        char** endptr, int base);
35        // char*(base) 轉 unsigned long long int
36        // 且指向轉換子字串之末
```

```
27 // Function: Integer arithmetics
28 int abs(int n);
29 long int labs(long int n);
30 long long int llabs(long long int n);
31 // Absolute value
32 }
```

2.2 algorithm

```
1 #include <algorithm>
2 using namespace std;
3 {
4     // FI(ForwardIterator)
5     // RAI(RandomAccessIterator)
6     // BI(BidirectionalIterator)
7     void sort(RAI first, RAI last);
8
9     FI lower_bound(FI first, FI last, const T& k);
10    /* 最左邊 ≥ k 的位置 */
11
12    FI upper_bound(FI first, FI last, const T& k);
13    /* 最左邊 > k 的位置 */
14
15    pair<FI,FI> equal_range(FI first, FI last, const
16        T& k);
17    /* 等於 k 的範圍 [lower_bound, upper_bound) */
18
19    bool next_permutation(BI first, BI last);
20    /* 使用已經排序(由小到大的)資料，產生下一組排列 */
21
22    bool prev_permutation(BI first, BI last);
23    /* 針對逆向排序(由大到小的)資料，產生上一組排序 */
24 }
```

2.3 map

2.4 set

2.5 vector

2.6 string

3 Algorithms

3.1 最短路

3.1.1 Bellman-Ford

3.1.2 Dijkstra's

3.2 LIS - Longest Increasing Subsequence

4 Formula

4.1 thm

• 中文測試

$$\sum_{i=1}^n i^2 = \frac{n(n+1)(2n+1)}{6}$$