在 Java 中分割字串, 可以用 **split** or **StringTokenizer** 兩種方式 :

 split 程式碼

String vData = "Hello World";

// 用哪個字元分割

String[] vElement = vData.**split**("o");

for( int i = 0; i < vElement.length; i++ )

System.out.println( vElement[i] );

// 結果

Hell

W

rld

 StringTokenizer 程式碼

String vData = "Hello World";

// 用哪個字元分割

java.util.**StringTokenizer** vElement = new java.util.**StringTokenizer**(vData, "o");

while( vElement.hasMoreTokens() )

out.println( vElement.nextToken() + "

" );

// 結果

Hell

W

rld

（Japanese）

今日は、HackerCupやCodeJam対策で学んだ便利なスキルについて。

標準入出力を指定したファイルに結びつける方法です。java風に言うと、標準入出力ストリームにファイルストリームを代入するみたいな感じです。

これを使うと、ファイル出力にprintf(), coutが使えます。ファイル入力にはscanf()、cinが使えます。かなり便利です。

C++、Javaそれぞれでサンプルコードを載せておきます。

INPUTファイルとOUTPUTファイルを作成してから、このソースを実行してください。INPUTファイルにはいくつかの整数を記入してください。

[C++ version]

view plainprint?

1. #define INPUT "C:/input.txt"
2. #define OUTPUT "C:/output.txt"
4. **int** main() {
5. // connect I/O streams to files
6. freopen(INPUT, "r", stdin);
7. freopen(OUTPUT, "w", stdout);
9. **int** x;
10. **while** (cin >> x) {
11. cout << x << endl;
12. }
14. cerr << "done." << endl;
15. **return** 0;
16. }

[Java version]

view plainprint?

1. **public** **class** Main {
2. **static** **private** final String INPUT = "C:/input.txt";
3. **static** **private** final String OUTPUT = "C:/output.txt";
5. **public** **static** **void** main(String args[]) {
6. // open I/O files
7. FileInputStream instream = null;
8. PrintStream outstream = null;
10. **try** {
11. instream = **new** FileInputStream(INPUT);
12. outstream = **new** PrintStream(**new** FileOutputStream(OUTPUT));
13. System.setIn(instream);
14. System.setOut(outstream);
15. } **catch** (Exception e) {
16. System.err.println("Error Occurred.");
17. }
19. Scanner in = **new** Scanner(System.in);
20. **for** (;in.hasNext();) {
21. **int** x = in.nextInt();
22. System.out.println(x);
23. }
25. System.err.println("done.");
26. **return**;
27. }
29. }