



AOJ

2417 – Flick Input

<http://judge.u-aizu.ac.jp/onlinejudge/description.jsp?id=2417>

Problem

1

1 あ行	2 か行	3 さ行
4 た行	5 な行	6 は行
7 ま行	8 や行	9 ら行
	0 わをん	

(a)

	上に フリック (う段)	
左に フリック (い段)	タッチ だけ (あ段)	右に フリック (え段)
	下に フリック (お段)	

(b)

	う			く			す	
い	あ	え	き	か	け	し	さ	せ
	お			こ			そ	
	つ			ぬ			ふ	
ち	た	て	に	な	ね	ひ	は	へ
	と			の			ほ	
	む			ゆ			る	
み	ま	め		や		り	ら	れ
	も			よ			ろ	
				ん				
				わ				
				を				

(c)

Problem

2

1 あ行	2 か行	3 さ行
4 た行	5 な行	6 は行
7 ま行	8 や行	9 ら行
	0 わをん	

(a)

	U	
L	T	R
	D	

(b)

	う			く			す	
い	あ	え	き	か	け	し	さ	せ
	お			こ			そ	
	つ			ぬ			ふ	
ち	た	て	に	な	ね	ひ	は	へ
	と			の			ほ	
	む			ゆ			る	
み	ま	め		や		り	ら	れ
	も			よ			ろ	
				ん				
				わ				
				を				

(c)

Problem

- ローマ字で出力

- 5R2D

- neko

- 8U9U6U0T

- yuruhuwa

- か行 : 'k'

- さ行 : 's'

- た行 : 't'

- な行 : 'n'

- は行 : 'h'

- ま行 : 'm'

- や行 : 'y'

- ら行 : 'r'

- わ行 : 'w'



- あ段 : 'a'

- い段 : 'b'

- う段 : 'c'

- え段 : 'd'

- お段 : 'e'

- ん : nn

解法

- やるだけ
- `std::map` が便利

std::map

```
map<string, double> gpa;  
gpa.insert(make_pair( "bowson" , 5.0));  
gpa.insert(make_pair( "morimo" , 6.0));  
cout << gpa[ "bowson" ] << endl;  
cout << gpa[ "morimo" ] << endl;  
---
```

```
$ ./a.out
```

```
5
```

```
6
```

std::map

```
map<string, double> gpa;  
gpa[ "bowson" ] = 5.0;  
gpa[ "morimo" ] = 6.0;  
cout << gpa[ "bowson" ] << endl;  
cout << gpa[ "morimo" ] << endl;  
---
```

```
$ ./a.out
```

```
5
```

```
6
```

std::unordered_map

参考

7

- C++11
 - `g++ -std=c++11 test.cpp`
- ハッシュ連想配列クラス `std::unordered_map`
 - ハッシュテーブルで実装
 - `std::map` は二分木
 - キーによるアクセス時間 $O(1)$
 - `std::map` は $O(\log N)$
 - 要素の順序付けは不可
 - `std::map` は順序付けされている

ソースコード

- [2417.cpp](#)

std::map

参考

9

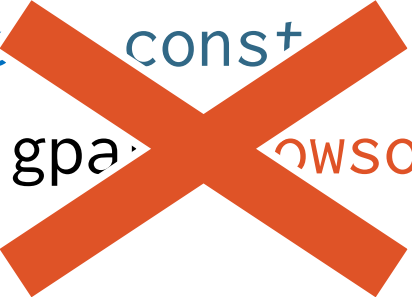
```
class Test {  
    private:  
        map<string, double> gpa;  
    public:  
        Test() {gpa[ "bowson" ] = 5.0;}  
        void print() const {  
            cout << gpa[ "bowson" ] << endl;  
        }  
};
```

std::map

参考

10

```
class Test {  
    private:  
        map<string, double> gpa;  
    public:  
        Test() {gpa[ "bowson" ] = 5.0;}  
        void print const  
            cout << gpa[ "bowson" ] << endl;  
        }  
};
```



std::map

参考

11

```
class Test {  
    :  
    void print() const {  
        map<string, double>::const_iterator pos;  
        pos = gpa.find( "bowson" );  
        cout << pos->second << endl;  
    }  
};
```

std::map

参考

12

```
class Test {  
    :  
    void print() const {  
        cout << gpa.find( "bowson" )->second <<  
            << endl;  
    }  
};
```

std::map (c++11)

参考

13

```
class Test {  
    :  
    void print() const {  
        cout << gpa.at( "bowson" ) << endl;  
    }  
};
```