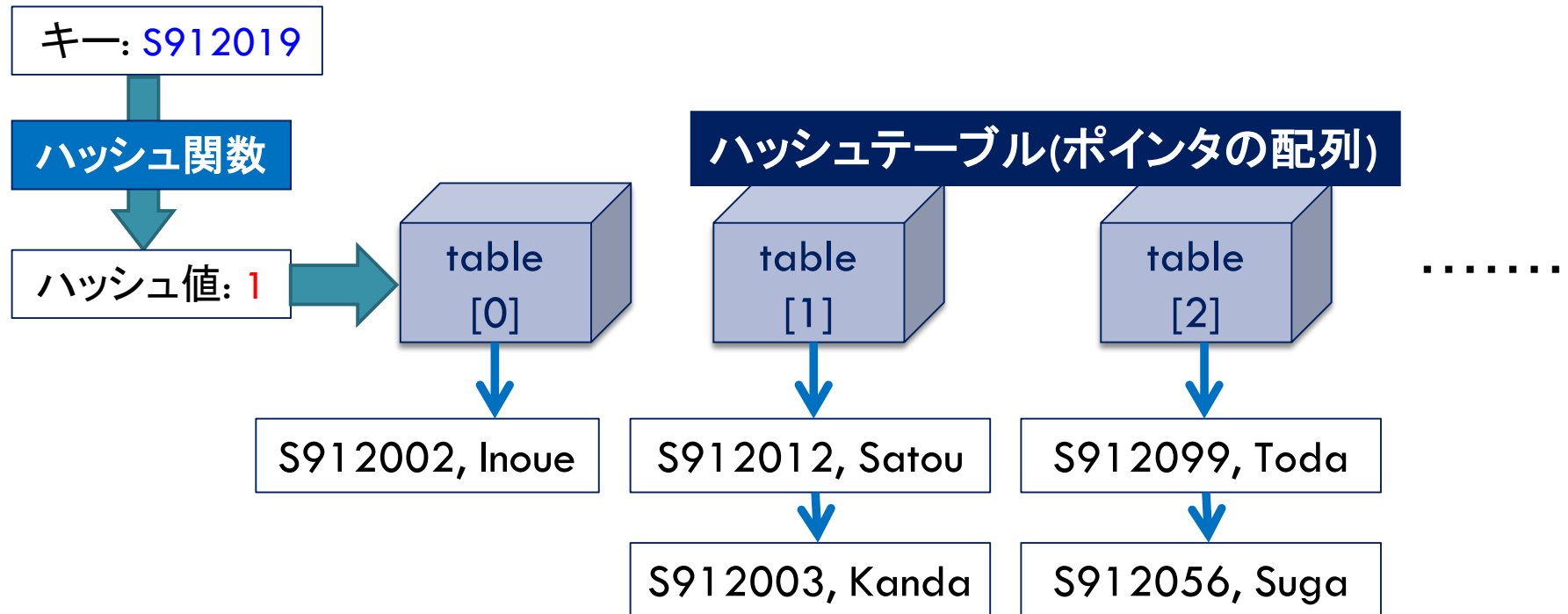


# 問題

第1引数にkey、第2引数にvalueをとり、  
すでにkeyとvalueの組み合わせが登録されていたら、0を返し、  
keyと別のvalueの組み合わせが登録されていたら、エラーを返し、  
keyが未登録であれば、ハッシュ値hashvalueにおけるリストの先頭ノードtable[hashvalue]に  
keyとvalueを格納した新しいノードを追加する関数

```
int Hash::insert(string key, string value)
```

を作成してください。HASHSIZEは7としてください。



# 問題(続き)

第1引数にkey、第2引数にvalueをとり、  
すでにkeyとvalueの組み合わせが登録されていたら、0を返し、  
keyと別のvalueの組み合わせが登録されていたら、エラーを返し、  
keyが未登録であれば、ハッシュ値hashvalueにおけるリストの先頭ノードtable[hashvalue]に  
keyとvalueを格納した新しいノードを追加する関数

```
int Hash::insert(string key, string value)
```

を作成してください。HASHSIZEは7としてください。

キー: S912019

ハッシュ関数

ハッシュ値: 1

ハッシュテーブル(ポインタの配列)

table  
[0]

S912002, Inoue

table  
[1]

S912019, Segi

S912012, Satou

S912003, Kanda

table  
[2]

S912099, Toda

S912056, Suga

.....

# 問題(続き)

//mainは以下のプログラムを使用してください。

```
int main(void){
    Hash ht(HASHSIZE);                                //ハッシュの作成
    string filename="data.csv";
    string line;
    ifstream fin(filename);
    if(!fin){                                           //エラー処理
        cerr << "エラー: ファイルを開けません" << filename << "\n";
        exit(EXIT_FAILURE);
    }
    while(fin >> line){
        int index = line.find(",");                    //カンマの添え字がindexに戻る
        string key = line.substr(0,index);              //lineの0番目からindex-1番目までの文字列をkeyにコピー
        string value = line.substr(index+1);           //lineのindex+1番目から終わりまでの文字列をvalueにコピー
        ht.insert(key,value);                          //ハッシュにkeyとvalueのペアを追加
    }
    fin.close();
    cout << "\n";
    ht.print();                                         //ハッシュの中身を出力

    string inputkey, inputvalue;
    cout << "\ninput key ? ";
    cin >> inputkey;
    cout << "input value ? ";
    cin >> inputvalue;
    if(ht.insert(inputkey, inputvalue)==1){            //inputkeyとinputvalueを追加。
        cout << "WARNING:input keyとinput valueはすでに登録されています\n";
    }else{
        ht.print();
    }
    return 0;
}
```

# 実行例1

```
comsv001% ./a.out
```

```
print Hash Table
```

```
0-> (S912002,Inoue)
```

```
1-> (S912012,Satou) -> (S912003,Kanda)
```

```
2-> (S912099,Toda) -> (S912056,Suga)
```

```
3-> (S912075,Taguchi)
```

```
4
```

```
5
```

```
6-> (S912001,Abe)
```

```
input key ? S912019
```

```
input value ? Segi
```

```
Print Hash Table
```

```
0-> (S912002,Inoue)
```

```
1-> (S912019,Segi) -> (S912012,Satou) -> (S912003,Kanda)
```

```
2-> (S912099,Toda) -> (S912056,Suga)
```

```
3-> (S912075,Taguchi)
```

```
4
```

```
5
```

```
6-> (S912001,Abe)
```

```
comsv001% ./a.out
```

```
print Hash Table
```

```
0-> (S912002,Inoue)
```

```
1-> (S912012,Satou) -> (S912003,Kanda)
```

```
2-> (S912099,Toda) -> (S912056,Suga)
```

```
3-> (S912075,Taguchi)
```

```
4
```

```
5
```

```
6-> (S912001,Abe)
```

```
input key ? S912002
```

```
input value ? Inoue
```

```
WARNING:input keyとinput valueはすでに登録されています
```

# 実行例2

comsv001% ./a.out

Print Hash Table

0-> (S912002,Inoue)

1-> (S912012,Satou) -> (S912003,Kanda)

2-> (S912099,Toda) -> (S912056,Suga)

3-> (S912075,Taguchi)

4

5

6-> (S912001,Abe)

input key ? S912002

input value ? Inohara

ERROR:key and different value have already been registered:S912002 Inohara Inoue

comsv001%