


## II) Programming

### 1. Hash Table (Separate Chaining)

 Microsoft Visual Studio 디버그 콘솔

First hash function  $h(k) = k \bmod 7$

```
735 833 532 945 483 252 490
757 1 813 484 904 813 680 176 743 106
660 576 436 737 310 310 324 226 534
521 717 535 493
592 39 319 984 396 389 424
138 502 635 68 292 306
811 293 118 524 951 97 174
```

Average chain length is 7  
Shortest chain length is 4  
Longest chain length is 10

Second hash function  $h(k) = k \bmod 13$

```
39
521 833 1 118 534
717 535 951 483 743 106
757 484 68
576 524 680
811 252 174 226
292 97 396
735 592 293 813 904 319 436 813 176 306
138 502 424
945 984 737 490
660
635 310 310
532 324 389 493
```

Average chain length is 3  
Shortest chain length is 1  
Longest chain length is 10

Third hash function  $h(k) = k \bmod 17$

```
833 68 680 493 306
1 324
138
717 904 292
735 293 310 310 106 174
39 532 396 226
635 737 176
483 534
484 535
757 502
945
521 436
811 97 743
319
592 660 813 524 813 252 490
576 984 389
118 951 424
```

```
Average chain length is 2
Shortest chain length is 1
Longest chain length is 7
```

체인이 비어 있다면, 빈 채로 두고, newline을 입력 후 진행하게 했습니다.

1번에는 2번과 같은 (note that you need a function for the 90 insertions, 30 for each table) 이라는 줄이 없어서, 30개의 randomly generated keys를 만든 후, 이 30개를 각 hash table에 넣었습니다.

2. Construct the open address hash table according to the following description.

Microsoft Visual Studio 디버그 콘솔

Linear Probing

```
1 : NULL
2 : NULL
3 : 594
4 : 39
5 : 337
6 : 853
7 : 744
8 : 669
9 : 489
10 : 7
11 : NULL
12 : 566
13 : 49
14 : 346
15 : 125
16 : 89
17 : 608
18 : 350
19 : 647
20 : 648
21 : 575
22 : 977
23 : NULL
24 : 763
25 : 912
26 : 173
27 : 359
28 : 249
29 : NULL
30 : 177
31 : 696
32 : 31
33 : 439
34 : NULL
35 : 441
36 : 997
37 : NULL
```

Primary cluster : 11  
Average number of probes : 1.60

Microsoft Visual Studio 디버그 콘솔

Quadratic Probing

```
1 : 148
2 : 445
3 : 261
4 : 595
5 : 670
6 : 116
7 : 561
8 : 377
9 : 337
10 : 9
11 : NULL
12 : 303
13 : 234
14 : 379
15 : 999
16 : 408
17 : NULL
18 : 128
19 : 277
20 : 130
21 : 57
22 : 646
23 : NULL
24 : NULL
25 : 205
26 : 428
27 : 359
28 : 730
29 : 768
30 : NULL
31 : 474
32 : NULL
33 : 916
34 : 447
35 : 330
36 : 146
37 : NULL
```

Primary cluster : 10  
Average number of probes : 1.47

## Double Hashing

```
1 : 97
2 : 922
3 : 247
4 : 259
5 : 27
6 : 989
7 : 80
8 : 363
9 : 855
10 : 194
11 : 670
12 : NULL
13 : NULL
14 : NULL
15 : 273
16 : 459
17 : 238
18 : 461
19 : 499
20 : 759
21 : 693
22 : 792
23 : 769
24 : 430
25 : NULL
26 : 284
27 : NULL
28 : 508
29 : NULL
30 : 991
31 : 30
32 : 397
33 : NULL
34 : 66
35 : 559
36 : 882
37 : 110
```

Primary cluster : 11

Average number of probes : 2.40

(i=0)일 때, slot이 비어 있어서 값을 넣은 경우에도 probing을 1회 한 것으로 계산하였습니다.

### 3. RBT (Red-Black Tree)

트리는 위에서부터 레벨별로 NULL Node를 포함해 모두 프린트하였습니다.

즉, 라인 별로 1, 2, 4, 8, 16...개의 항목을 표시하게 됩니다.

#### Microsoft Visual Studio 디버그 콘솔

```
After inserting random 20 keys
19[B]
14[B] 33[B]
5[R] 17[B] 30[R] 35[R]
3[B] 11[B] 15[R] 18[R] 27[B] 32[B] 34[B] 40[B]
0[R] N N 12[R] N N N N N N 31[R] N N N 37[R] 47[R]

After inserting 2
19[B]
14[B] 33[B]
5[R] 17[B] 30[R] 35[R]
2[B] 11[B] 15[R] 18[R] 27[B] 32[B] 34[B] 40[B]
0[R] 3[R] N 12[R] N N N N N N 31[R] N N N 37[R] 47[R]

After inserting 22
19[B]
14[B] 33[B]
5[R] 17[B] 30[R] 35[R]
2[B] 11[B] 15[R] 18[R] 27[B] 32[B] 34[B] 40[B]
0[R] 3[R] N 12[R] N N N N 22[R] N 31[R] N N N 37[R] 47[R]

After inserting 13
19[B]
14[B] 33[B]
5[R] 17[B] 30[R] 35[R]
2[B] 12[B] 15[R] 18[R] 27[B] 32[B] 34[B] 40[B]
0[R] 3[R] 11[R] 13[R] N N N N 22[R] N 31[R] N N N 37[R] 47[R]
Already present

After inserting 47
19[B]
14[B] 33[B]
5[R] 17[B] 30[R] 35[R]
2[B] 12[B] 15[R] 18[R] 27[B] 32[B] 34[B] 40[B]
0[R] 3[R] 11[R] 13[R] N N N N 22[R] N 31[R] N N N 37[R] 47[R]
```

```
After inserting 36
19[B]
14[B] 33[R]
5[R] 17[B] 30[B] 35[B]
2[B] 12[B] 15[R] 18[R] 27[B] 32[B] 34[B] 40[R]
0[R] 3[R] 11[R] 13[R] N N N N 22[R] N 31[R] N N N 37[B] 47[B]
N N N N N N N N N N N N N N N N N N N N N N N 36[R] N N N
```

Deletion은 구현해내지 못했습니다.