

```
// 「Cで学ぶデータ構造とアルゴリズム」(西原清一)オーム社, 2008
// 図8・5(p.190) 2次走査法による開番地ハッシュ法
// 改訂版
```

```
#include<stdio.h>
#include<string.h>
#define M 11
#define NON 0

int tab[M];

int hash0(int K) {return K%M;}

void quadratic_store(int K, int a, int b)
{
    int h,u,v;
    h = hash0(K);
    u = a+b; v = 2*b;
    while (tab[h] != NON) {h = (h+u)%M; u = u+v;}
    tab[h] = K;
}

int quadratic_search(int K, int a, int b)
{
    int h,u,v;
    h = hash0(K);
    u = a+b; v = 2*b;
    while (tab[h] != NON)
        if (K == tab[h]) return h;
        else {h = (h+u)%M; u = u+v;}
    return -1;
}

main()
{
    int s,K,i,h,a=1,b=1;
    for (i=0; i<M; i++) tab[i]=NON;
    printf("input s(1:add, -1:search, 0:stop), and key K(>0)--- OK?¥n");
    do {
        printf("s? K? ");
        scanf("%d%d",&s,&K);
        switch (s) {
            case 1: quadratic_store(K,a,b);
```

```

                                break;
        case -1:    h = quadratic_search(K, a, b);
                                printf("    *found at address %d\n", h);
                                break;
        default:    break;
    }
}

while (s != 0);

for (i=0; i<M; i++) printf("%d=%d  ", i, tab[i]);
printf("\n");
}

```