Hash_main_char.c

```
/*
 *@Title:   ハッシュ法の動作検証用main関数(キーの型: 文字列)
 *@Author:  後藤 智範
 *@Date:    2013.10.2
 */

#include "Hash_main_char.h"                 // ハッシュ法メイン関数用ヘッダーファイル
#include "measurement.h"
// 次の値は集合の濃度(要素数)に応じて設定する
#define  HSIZE      100000                  // ハッシュ表の大きさ(10万)

//#define  DVS         99991                 // 除数(HSIZEを超えない最大の素数)
//#define  HSIZE       10000                 // ハッシュ表の大きさ(10万)
#define  DVS          9973                  // 除数(HSIZEを超えない最大の素数)
#define  CONSL          19                  // 線形探査法の定数

// 次の4種類のハッシュ法のいずれかのコメントをはずす
#include "Hash_open_char.h"       // 開放番地法
//#include "Hash_chain_char.h"     // 連鎖法
//#include "Hash_externa_char.h"   // 外部ハッシュ法
//#include "Hash_extend_char.h"    // 拡張ハッシュ法



int main(int argc, char *argv[]) {
  FILE *fin ;
  int     cmdnum ,                      // コマンド番号
    rc ,                          // 関数の戻り値
    itemno = 0 ,                  // 処理要素数
    oun = HSIZE / 10 ;            // 処理レコード数出力単位
  char  msg[] = "Usage: Hash_main.exe 入力ファイル(整数値の集合) DVS CONSL" ;
```

```
  char  inv[LSTR] ;                          // 入力値(集合の要素)
  int j;
  double msec = 0;
  // 次の2行は、拡張ハッシュ法で必要な宣言
  //      DIRECTORY *directory ;             //  ディレクトリー
  //      int        depth = IDEPTH ;        // ディレクトリーの次数の
初期値

  fin = open_file(FILENAME, "r") ;           // 入力ファイル(整数の集合)

// 次の2行は、拡張ハッシュ法以外で必要な処理
  Htable = allocate_Htable(HSIZE) ;      // ハッシュ表の動的割付け
  init_tbl(Htable, HSIZE) ;              // ハッシュ表初期化

  // 以下の行は、拡張ハッシュ法で必要な処理
  //      directory = create_directory(depth) ;
  //      inint_directory(directory, depth) ;
  //      for(i = 0; i < (int)pow(2, depth); i++)
  //      directory[i].ref = create_bucket(depth);
  //*

  printf("要素数¥t衝突番地総数¥t総探査回数¥t格納時間¥n");
  while(fscanf(fin, FMTF, inv) != EOF) {  // キーの読込み
    STORE ;  // キーの格納
    if (rc == FLS)        printf("%s は格納できません¥n", inv);
    if (++itemno % oun == 0){
      SYAD = 0;
      for(j = 0; j < HSIZE; j++){
        if(Table[j] > 1){
          Syad++;
        }
      }
      printf("%dU¥t%d¥t%d¥t%d¥n", itemno/oun, Syad, Frq_prob);
    }
  }
```

```
  fclose(fin) ;                                    // ファイル・クローズ
  return ;
}
```