

Apr 23, 04 12:01

ens.h

Page 1/1

```

#ifndef ENS_H
#define ENS_H

/* MULTI-ENSEMBLE D'ENTIERS.
   Un même entier peut être présent plusieurs fois. */

/* Doit être utilisée avant tout usage du module ens. */
void
ens_initialiser(void);

/* Ajoute un élément.
   Provoque une erreur si l'ensemble est plein. */
void
ens_ajouter(int element);

/* Supprime un élément.
   Provoque une erreur si l'élément n'est pas présent. */
void
ens_supprimer(int element);

/* Teste la présence d'un élément dans l'ensemble. */
int
ens_present(int element);

#endif

```

Apr 26, 04 12:40

ens.c

Page

```

#include <assert.h>
#include "ens.h"

#define TAILLE 10

/* contient les éléments de l'ensemble */
static int ens[TAILLE];
/* précise si l'indice correspondant de ens est disponible
   pour accueillir un nouvel élément. */
static int vide[TAILLE];

void
ens_initialiser(void)
{
    int i;

    for (i = 0; i < TAILLE; i++)
        vide[i] = 1;
}

static int
premier_vide()
{
    int i;
    for (i = 0; i < TAILLE && !vide[i]; i++)
        ;
    return i;
}

void
ens_ajouter(int element)
{
    int i = premier_vide();
    assert(i < TAILLE);
    ens[i] = element;
    vide[i] = 0;
}

/* Retourne l'indice de la première occurrence d'un element,
   TAILLE si cet élément n'est pas présent.
*/
static int
premiere_occurrence(int element)
{
    int i;

    for (i = 0; i < TAILLE; i++)
        if (! vide[i] && ens[i] == element)
            return i;
    return TAILLE;
}

void
ens_supprimer(int element)
{
    int position = premiere_occurrence(element);

    assert(position < TAILLE);
    vide[position] = 1;
}

```