

TP8_INF203

Groupe: INF2

Binôme: Shaghayegh HAJMOHAMMADKASHI

Kaiwen ZHENG

Terminal : cd INF203

./TP1/scripts/installeTP.sh 8

cd TP8

[a] :

Le nom du type permettant de représenter un ensemble de joueurs est "joueurs".

Les joueurs dans un tel ensemble sont représentés sous forme d'une structure "joueur", qui contient un champ pour le pseudo du joueur (une chaîne de caractères) et un champ pour le nombre de billes du joueur (un entier).

Le cardinal maximal d'un tel ensemble est défini par la constante "NB_MAX_JOUEURS", qui vaut 100.

Oui, il y a une limite à la taille du nom d'un joueur, qui est définie par la constante "TAILLE_MAX_NOM", qui vaut 32.

[b] :

```
hajmohas@im2ag-turing-01:[~/INF203/TP8]: gcc -o billes billes.c joueurs.c generer_entier.c
```

```
hajmohas@im2ag-turing-01:[~/INF203/TP8]: ./billes Atchoum Dormeur Prof Atchoum
```

Copie successive des arguments dans l'ensemble :

Atchoum - Dormeur - Prof - Atchoum -

Ensemble de joueurs dans lequel la recherche est faite :

{ }

Qui voulez-vous ? Simplet

-- Joueur absent de l'ensemble

Qui voulez-vous ? Atchoum

-- Joueur absent de l'ensemble

Qui voulez-vous ? q

[c] :

```
#include <string.h>
#include <stdio.h>
#include "joueurs.h"

void init_joueurs(joueurs *ens) {
    ens->nb = 0;
```

```

}

int nombre_joueurs(joueurs *ens) {
    return ens->nb;
}

char *nom_joueur(joueurs *ens, int i) {
    return ens->T[i].pseudo;
}

int billes_joueur(joueurs *ens, int i) {
    return ens->T[i].nb_billes;
}

int trouver_joueur(joueurs *ens, char *nom) {
    int i;
    for (i = 0; i < nombre_joueurs(ens); i++) {
        if (strcmp(ens->T[i].pseudo, nom) == 0) {
            return i;
        }
    }
    return -1;
}

int ajouter_joueur(joueurs *ens, char *nom, int billes) {
    if (nombre_joueurs(ens) >= NB_MAX_JOUEURS) {
        return -1; // ensemble plein
    }
    if (trouver_joueur(ens, nom) != -1) {
        return -1; // le joueur est déjà présent dans l'ensemble
    }
    strcpy(ens->T[nombre_joueurs(ens)].pseudo, nom);
    ens->T[nombre_joueurs(ens)].nb_billes = billes;
    ens->nb++;
    return 0; // ajout réussi
}

void modifier_billes(joueurs *ens, int i, int cpt) {
    if (ens != NULL && i >= 0 && i < nombre_joueurs(ens)) {
        ens->T[i].nb_billes = cpt;
    }
}

void afficher_ensemble_joueurs(joueurs *ens) {
    int i;
    for (i=0 ; i<nombre_joueurs(ens) ; i++)
        printf("%s %d\n", ens->T[i].pseudo, ens->T[i].nb_billes) ;
}

```

[d] :

```
#include <string.h>
#include <stdio.h>
#include "joueurs_out.h"

void ecrire_les_joueurs(joueurs *ens, char *nom_fich) {
    FILE *fichier = fopen(nom_fich, "w");
    fprintf(fichier, "%d\n", nombre_joueurs(ens));
    for (int i = 0; i < nombre_joueurs(ens) ; i++){
        fprintf(fichier, "%s %d\n", nom_joueur(ens, i), billes_joueur(ens,
i));
    }
    fclose(fichier);
}
```

[e] :

```
hajmohas@im2ag-turing-01:[~/INF203/TP8]: gcc -o billes billes.c joueurs.c generer_entier.c
joueurs_out.c
```

```
hajmohas@im2ag-turing-01:[~/INF203/TP8]: ./billes Atchoum Dormeur Prof Atchoum
```

Copie successive des arguments dans l'ensemble :

Atchoum - Dormeur - Prof - Atchoum -

Ensemble de joueurs dans lequel la recherche est faite :

{ Atchoum 135 Dormeur 28 Prof 2 }

Qui voulez-vous ? Atchoum

135

Qui voulez-vous ? q

```
hajmohas@im2ag-turing-01:[~/INF203/TP8]: cat ./Fichier_Joueurs
```

3

Atchoum 135

Dormeur 28

Prof 2

[f] :

joueurs_in.h

```
#ifndef JOUEURS_IN_H
#define JOUEURS_IN_H
#include <stdio.h>
#include <stdlib.h>
#include "joueurs.h"
```

```
void lire_les_joueurs(joueurs *ens, char *nom_fich);

#endif
```

joueurs_in.c

```
#include <stdio.h>
#include <stdlib.h>
#include "joueurs_in.h"

void lire_les_joueurs(joueurs *ens, char *nom_fich) {
    FILE *fichier = fopen(nom_fich, "r");
    if (fichier == NULL) {
        fprintf(stderr, "Erreur lors de l'ouverture du fichier %s\n",
nom_fich);
        exit(1);
    }
    int nb_joueurs;
    fscanf(fichier, "%d\n", &nb_joueurs);
    char nom[TAILLE_MAX_NOM];
    int nb_billes;
    for (int i = 0; i < nb_joueurs ; i++){
        fscanf(fichier, "%s %d\n", nom, &nb_billes);
        ajouter_joueur(ens, nom, nb_billes);
    }
    fclose(fichier);
}
```

Billes.c

```
#include <stdio.h>
#include <string.h>
#include "generer_entier.h"
#include "joueurs.h"
#include "joueurs_out.h"
#include "joueurs_in.h"

void afficher_joueur(joueur* pj) {
    printf("%s a %d billes\n", pj->pseudo, pj->nb_billes) ;
}

void lire_joueur(joueur* pj) {
    printf("Son nom ? ");
    scanf("%s", pj->pseudo);
    printf("Combien de billes ? ");
    scanf("%d", &pj->nb_billes);
}
```

```

}

int main(int argc, char *argv[]) {
    int i;
    joueurs ens_joueurs;
    char nom[TAILLE_MAX_NOM] = "";

    init_joueurs(&ens_joueurs);

    if (argc > 1) {
        printf("Copie successive des arguments dans l'ensemble :\n");
        for (i = 1; i < argc; i++) {
            printf(" %s -", argv[i]);
            ajouter_joueur(&ens_joueurs, argv[i], generer_entier(500));
        }
        printf("\n");
    } else {
        lire_les_joueurs(&ens_joueurs, "Fichier_Joueurs");
    }

    printf("Ensemble de joueurs dans lequel la recherche est faite :\n");
    /***** ou appel de la fct afficher_ensemble_joueurs *****/
    printf("{ ");
    for (i=0; i<nombre_joueurs(&ens_joueurs); i++) {
        printf("%s %d ", nom_joueur(&ens_joueurs, i),
            billes_joueur(&ens_joueurs, i));
    }
    printf("}\n");

    printf("Qui voulez-vous ? ");
    scanf("%s", nom);
    while ( strcmp(nom, "q") != 0) {
        i = trouver_joueur(&ens_joueurs, nom);
        if (i == -1) {
            printf("-- Joueur absent de l'ensemble\n");
        } else {
            printf("%d\n", billes_joueur(&ens_joueurs, i));
        }
        printf("Qui voulez-vous ? ");
        scanf("%s", nom);
    }
    ecrire_les_joueurs(&ens_joueurs, "Fichier_Joueurs");
    return 0;
}

```

[g] :

```
void modifier_billes(joueurs *ens, int i, int cpt) {
    if (ens != NULL && i >= 0 && i < nombre_joueurs(ens)) {
        ens->T[i].nb_billes = cpt;
    }
}
```

Operations.c

```
#include <stdio.h>

#include "joueurs.h"
#include "operations.h"

int gain(joueurs *e) {
    char nom[TAILLE_MAX_NOM] ;
    int i, solde, depose ;

    printf("Nom du joueur et billes gagnées ? ") ;
    scanf("%s %d", nom, &depose) ;
    i = trouver_joueur(e, nom) ;
    if (i == -1) {
        printf("Inconnu\n") ; return 1 ;
    }
    solde = billes_joueur(e, i) ;
    solde = solde + depose ;
    modifier_billes(e, i, solde) ;
    return 0 ;
}

int perte(joueurs *e) {
    char nom[TAILLE_MAX_NOM] ;
    int i, solde, retire ;

    printf("Nom du joueur et billes perdues ? ") ;
    scanf("%s %d", nom, &retire) ;
    i = trouver_joueur(e, nom) ;
    if (i == -1) {
        printf("Inconnu\n") ; return 1 ;
    }
    solde = billes_joueur(e, i) ;
    solde = solde - retire ;
    modifier_billes(e, i, solde) ;
    return 0 ;
}

/* Lecture du nom du joueur a debiter, du nom
du joueur a crediter, et du montant du transfert.
```

```

Mise a jour des deux comptes.*/
int transfert(joueurs *e) {
    char nom_debiteur[TAILLE_MAX_NOM], nom_crediteur[TAILLE_MAX_NOM];
    int i_debiteur, i_crediteur, montant;

    printf("Nom du joueur à débiter : ");
    scanf("%s", nom_debiteur);
    i_debiteur = trouver_joueur(e, nom_debiteur);
    if (i_debiteur == -1) {
        printf("Joueur inconnu\n");
        return 1;
    }

    printf("Nom du joueur à créditer : ");
    scanf("%s", nom_crediteur);
    i_crediteur = trouver_joueur(e, nom_crediteur);
    if (i_crediteur == -1) {
        printf("Joueur inconnu\n");
        return 1;
    }

    printf("Montant à transférer : ");
    scanf("%d", &montant);
    if (montant <= 0) {
        printf("Le montant doit être positif\n");
        return 1;
    }
    if (billes_joueur(e, i_debiteur) < montant) {
        printf("%s n'a pas suffisamment de billes pour effectuer le
transfert\n", nom_debiteur);
        return 1;
    }

    modifier_billes(e, i_debiteur, billes_joueur(e, i_debiteur) - montant);
    modifier_billes(e, i_crediteur, billes_joueur(e, i_crediteur) + montant);

    return 0;
}

// int main() {
//     joueurs ens;
//     init_joueurs(&ens);

//     // Ajouter des joueurs
//     ajouter_joueur(&ens, "Alice", 10);
//     ajouter_joueur(&ens, "Bob", 20);
//     ajouter_joueur(&ens, "Charlie", 30);

```

```

//      // Afficher les joueurs
//      afficher_ensemble_joueurs(&ens);

//      // Test de la fonction gain
//      printf("\nTest de la fonction gain :\n");
//      gain(&ens);
//      afficher_ensemble_joueurs(&ens);

//      // Test de la fonction perte
//      printf("\nTest de la fonction perte :\n");
//      perte(&ens);
//      afficher_ensemble_joueurs(&ens);

//      // Test de la fonction transfert
//      printf("\nTest de la fonction transfert :\n");
//      transfert(&ens);
//      afficher_ensemble_joueurs(&ens);

//      return 0;
//  }

```

Gestion.c

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#include "joueurs_in.h"
#include "joueurs_out.h"
#include "operations.h"

int main (int argc, char *argv[]) {
    joueurs ens;
    init_joueurs(&ens); // j'ajoute ça
    char requete[9] ; char rqt ;
    int fini=0 ;
    if (argc != 3) {
        printf("syntaxe : %s fich_joueurs_entree fich_joueurs_sortie\n",
argv[0]) ;
        exit(1) ;
    }
    lire_les_joueurs(&ens, argv[1]) ;
    afficher_ensemble_joueurs(&ens) ;
    while (!fini) {
        printf("\n Requête (G,P,T,Q) : ") ;
        scanf("%s", requete);
        rqt = requete[0] ;
        switch (rqt) {
            case 'G' : gain(&ens) ; break ;

```



```

        case 'P' : perte(&ens) ; break ;
        case 'T' : transfert(&ens) ; break ;
        case 'Q' : fini = 1 ;
        default : ;
    }
}
afficher_ensemble_joueurs(&ens) ;
ecrire_les_joueurs(&ens, argv[2]) ;
return 0 ;
}

```

hajmohas@im2ag-turing-01:[~/INF203/TP8]: gcc -o gestion gestion.c joueurs_in.c joueurs_out.c operations.c joueurs.c

hajmohas@im2ag-turing-01:[~/INF203/TP8]: ./gestion Fichier_Joueurs Nouveau_Fichier

Atchoum 310

Dormeur 226

Prof 193

Requête (G,P,T,Q) : G

Nom du joueur et billes gagnées ? Atchoum

5

Requête (G,P,T,Q) : P

Nom du joueur et billes perdues ? Prof

3

Requête (G,P,T,Q) : T

Nom du joueur à débiter : Dormeur

Nom du joueur à créditer : Prof

Montant à transférer : 6

Requête (G,P,T,Q) : Q

Atchoum 315

Dormeur 220

