

```

1  /* #####
2  *» » » » » » » » semaine.h
3  *
4  *
5  * Fichier contenant Les déclarations relatives à La gestion de La liste
6  * semaines
7  *
8  * #####
9
10
11 #ifndef __SEMAINE_H_GUILLAUME_FARAH__
12 #define __SEMAINE_H_GUILLAUME_FARAH__
13
14 #include "actions.h"
15 #include <stdlib.h>
16 #include <string.h>
17 #include <stdio.h>
18
19
20 typedef struct semaines {
21     » char * semaine;
22     » struct actions * actions;
23     » struct semaines * semaineSuivante;
24 }semaines_t;
25
26 #define MAXLISTE 19
27
28
29 /*-----
30 *» » » » » » createSemaine
31 *
32 * » » Procédure qui parcourt la liste des semaines et ajoute une semaine
33 * nécessaire) dans l'ordre en fonction de la chaîne de caractère complè
34 * (ex:
35 * 201720114Medecin) puis ajoute l'action correspondant avec la fonction
36 * createAction()
37 *
38 * Entrées
39 * » » semaines : pointeur de la liste des semaines
40 * » » chaîne : chaîne de caractère correspondant aux informations d'une
41 *
42 * -----
43 void createSemaine(semaines_t *, char *);
44
45 /*-----
46 * » » » » » alloueSemaine
47 *
48 * » » Fonction qui alloue un élément semaine sur le poiteur passé en
49 * paramètre et
50 * ajoute le champ semaine
51 *
52 * Entrées
53 * » » ptrSemaine : Pointeur de semaine sur lequel il faut faire l'allocation
54 * » » nomSemaine : Nom de la semaine à allouer
55 *

```

```

• -----
53 void alloueSemaine(semaines_t *, char[]);↵
54 ↵
55 /*-----
56  * » » » LireFichier↵
57  *↵
58  * » » Fonction qui lit un fichier donné en paramètre et créer Les semaines
59  * correspondantes↵
60  *↵
61  * Entrée↵
62  * » » nomFichier : nom du fichier à lire↵
63  *↵
64  * Sortie↵
65  * » » retourne le pointeur de la liste des semaines créée↵
66  * -----
67 semaines_t * lireFichier(char *);↵
68 ↵
69 /*-----
• -↵
70  * » » » » recherche↵
71  *↵
72  * » » Permet de parcourir la liste des semaines et de trouver la semaine
73  * » correspondante à l'élément recherché↵
74  *↵
75  * Entrées↵
76  * » semaines : pointeur de la liste des semaines à parcourir↵
77  * » semaine : chaîne de caractère correspondant à la semaine à trouver↵
78  *↵
79  * Sortie↵
80  * » Pointeur de pointeur de l'élément trouvé (peut-être pointeur sur un é
81  * » NIL ↵
82  *
• -----
83 semaines_t ** recherche(semaines_t *, char *);↵
84 ↵
85 /*-----
86  * » » » » sauvegardeSemaine↵
87  *↵
88  * » » Fonction qui permet de sauvegarder la structure en parcourant toute
89  * » les semaines et appelant la fonction de sauvegarde sur la liste des
90  * » actions↵
91  *↵
92  * » Entrées↵
93  * » » listeSemaine : pointeur sur la liste des semaines à sauvegarder↵
94  * » » nomFichier : nom du fichier où sauvegarder↵
95  * -----*,
96 void sauvegardeSemaine(semaines_t *, char *);↵
97 ↵
98 /*-----
99  * » » » » createListeJourFromActionSemaine↵
100  *↵
101  * » Fonction qui crée la liste des jours contenant une action donnée en
102  * » parcourant la liste des semaines↵
103  *↵
104  * » Entrées↵

```

```

105  /* » ListeSemaine : pointeur de la liste des semaines à parcourir
106  /* » action : chaîne de caractère à rechercher
107  /*
108  /* » Sortie
109  /* » Chaîne de caractère correspondant à la Liste des jours résultants
110  /* -----
111  char * createListeJourFromActionsSemaine(semaines_t *, char *);
112
113  /*-----
114  /* » » » » » supprimeActionInSemaines
115  /*
116  /* » » Supprime une action de la sdd à partir de l'année, la semaine, le .
117  /* » ainsi que l'heure en recherchant la semaine correspondante et appelle
118  /* » fonction de suppression sur une liste d'action
119  /*
120  /* » Entrées
121  /* » ListeSemaine : pointeur de la liste des semaines à parcourir
122  /* » chaîne : chaîne contenant toutes les informations pour supprimer

```