

```

1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <sys/types.h>
4 #include <sys/stat.h>
5 #include <fcntl.h>
6 #include <dirent.h>
7 #include <string.h>
8 #include <unistd.h>
9 #include <string.h>
10 #include "create.h"
11 #include "extract.h"
12 #include "parser.h"
13 #include "encryption.h"
14
15 /*
16 *   main
17 *   -----
18 *   Metodo principal de la aplicacion mytar
19 *
20 *   argc: Numero de argumentos suministrados.
21 *   argv: Argumentos suministrados.
22 *
23 *   Retorno: 0 en caso de ejecucion correcta. Numero negativo en caso de error.
24 */
25 int main(int argc, char** argv) {
26     /*Eetero para verificar el retorno de las funciones*/
27     int i;
28     /*Se crear la estructura contenedora de las opciones de mytar*/
29     mytar_instructions *instructions = parse(argc, argv);
30     /*Error de creacion*/
31     if (!instructions){
32         printf("Can't parse the mytar options or its arguments\n");
33         return 0;
34     }
35     /*Verifica si las opciones son validas*/
36     if(verifyOptions(*instructions) == -1){
37         printf("The mytar options are incorrect\n");
38         return 0;
39     }
40
41     /*Se realiza la creacion del .mytar con las opciones y argumentos dados*/
42     if (instructions->mytar_options[C]){
43         i = createMyTar(instructions->num_args, instructions->creation_directory,
44 *instructions);
45
46         if (i == -1){
47             printf("The creation was stoped because execution errors\n");
48             return 0;
49         }
50
51     /*
52     Se realiza la extracion o muestra del .mytar con las opciones y argumentos
53     dados
54     */
55     if (instructions->mytar_options[X] || instructions->mytar_options[T]){
56         i = extractMyTar(instructions->creation_directory, *instructions);
57
58         if (i == -1){
59             if (instructions->mytar_options[T]){

```

```
60         printf("The display was stoped because execution errors\n");
61     } else {
62         printf("The extraction was stoped because execution errors\n");
63     }
64     return 0;
65 }
66 }
67
68     free(instructions);
69
70     return 0;
71 }
72
```