UNIVERSITÉ DE PAU ET DES PAYS DE L'ADOUR | COLLÈGE STEE SCIENCES ET TECHNOLOGIES POUR L'ÉNERGIE ET L'ENVIRONNEMENT

Techniques de programmation - Projet - VRSP

Samson Pierre <samson.pierre@univ-pau.fr>

27/11/2018

Ce projet est un programme nommé VRSP (Video Rental Shop Program) qui doit lire un fichier XML dont le chemin est passé en ligne de commande. Ce fichier XML contient des données concernant un vidéo-club et les films qu'il loue. Ce programme doit proposer de saisir des commandes pour afficher des informations contenues dans ce fichier XML.

Le travail à rendre

L'archive à rendre doit contenir exactement ces fichiers :

```
$ tar tf projet-khaled-khebbeb-samson-pierre.tar.xz | sort
projet-khaled-khebbeb-samson-pierre/
projet-khaled-khebbeb-samson-pierre/Makefile
projet-khaled-khebbeb-samson-pierre/movie.c
projet-khaled-khebbeb-samson-pierre/movie.h
projet-khaled-khebbeb-samson-pierre/vrs.c
projet-khaled-khebbeb-samson-pierre/vrs.h
projet-khaled-khebbeb-samson-pierre/vrsp.c
projet-khaled-khebbeb-samson-pierre/vrs.xml
$
```

Les prénoms `khaled` et `samson` ainsi que les noms `khebbeb` et `pierre` sont à remplacer par les vôtres. Le fichier `Makefile` contient les règles dont se sert le programme `make`. Les fichiers `movie.c` et `movie.h` contiennent respectivement les définitions et les déclarations pour un film. Les fichiers `vrs.c` et `vrs.h` contiennent respectivement les définitions et les déclarations pour un vidéo-club. Le fichier `vrsp.c` contient les définitions pour le programme (notamment celle de la fonction principale). Le fichier `vrs.xml` contient des données concernant un vidéo-club et les films qu'il loue.

La compilation du projet doit se passer ainsi :

```
$ make
gcc -std=c89 -pedantic -Wall -Werror -g `pkg-config libxml-2.0 --cflags` -c -o movie.o movie.c
gcc -std=c89 -pedantic -Wall -Werror -g `pkg-config libxml-2.0 --cflags` -c -o vrs.o vrs.c
gcc -std=c89 -pedantic -Wall -Werror -g `pkg-config libxml-2.0 --cflags` -o vrsp.out vrsp.c vrs.o
movie.o `pkg-config libxml-2.0 --libs-only-L` `pkg-config libxml-2.0 --libs-only-l`
$
```

Vous êtes autorisés à utiliser les fonctions suivantes :

```
char *fgets(char *s, int size, FILE *stream);
int fprintf(FILE *stream, const char *format, ...);
void free(void *ptr);
int getchar(void);
void *malloc(size_t size);
int printf(const char *format, ...);
void *realloc(void *ptr, size_t size);
int strcmp(const char *s1, const char *s2);
size_t strlen(const char *s);
char *strstr(const char *haystack, const char *needle);
double strtod(const char *nptr, char **endptr);
long int strtol(const char *nptr, char **endptr, int base);
void xmlCleanupParser(void);
xmlNodePtr xmlDocGetRootElement(const xmlDoc *doc);
void xmlFreeDoc(xmlDocPtr cur);
xmlChar *xmlGetProp(const xmlNode *node, const xmlChar *name);
xmlChar *xmlNodeListGetString(xmlDocPtr doc, const xmlNode *list, int inLine);
xmlDocPtr xmlParseFile(const char *filename);
int xmlStrcmp(const xmlChar *str1, const xmlChar *str2);
```

Pour obtenir une bonne note, vous devez respecter ces règles :

— les fichiers doivent être encodés en UTF-8
— les fichiers ne doivent pas contenir de fautes d'orthographe
— les noms de fichiers doivent être ceux indiqués dans ce sujet

— les fichiers d'en-tête doivent être identiques à ceux du sujet
— le fichier XML doit être identique à celui du sujet
— les options de compilation doivent être celles précisées dans ce sujet
— les affichages à l'écran doivent correspondre à ceux du sujet
— la solution doit se rapprocher au maximum de ce qui est demandé dans ce sujet
— le code doit être correctement indenté
— le code doit être homogène concernant le nom des variables, les espaces, ...
— la mémoire allouée doit être correctement libérée
— les fonctions que vous utilisez dans votre code doivent figurer parmi celles autorisées dans ce sujet
— les valeurs de retour des fonctions pouvant échouer doivent être vérifiées afin de traiter les erreurs
— les messages d'erreur doivent être envoyés dans le flux d'erreur standard
— les autres messages seront envoyés dans le flux de sortie standard
— le code retourné par un processus rencontrant une erreur doit être 1
— le code retourné par un processus se terminant normalement doit être 0
— l'archive ne doit pas contenir d'autres fichiers que ceux demandés dans ce sujet
— l'archive doit être envoyée au plus tard le 14/12/2018 à 23:59
— l'archive doit être envoyée par e-mail à l'adresse `khaled.khebbeb@univ-pau.fr` pour le groupe A
— l'archive doit être envoyée par e-mail à l'adresse `samson.pierre@univ-pau.fr` pour les groupes B et C
— le sujet de l'e-mail doit être `TDP - Projet - Khaled Khebbeb - Samson Pierre`
— les prénoms `Khaled` et `Samson` sont à remplacer par les vôtres
— les noms `Khebbeb` et `Pierre` sont à remplacer par les vôtres
— le travail est à réaliser impérativement en binôme

Le programme

Le programme doit lire un fichier XML dont le chemin est passé en ligne de commande. Si le nombre de paramètres passés en ligne de commande est différent de un, le programme doit afficher un message d'erreur et quitter. Idem si l'analyse du fichier échoue (exemple : s'il n'existe pas).

```
$ ./vrsp.out
./vrsp.out: Invalid number of arguments
$ ./vrsp.out file.xml
I/O warning : failed to load external entity "file.xml"
./vrsp.out: Unable to parse the document
$
```

Le programme doit proposer de saisir des commandes :

```
$ ./vrsp.out vrs.xml
VRSP>
```

La commande `help` permet d'obtenir cet affichage :

```
$ ./vrsp.out vrs.xml
VRSP> help
addr: Prints the VRS address
help: Prints this help
mv: Prints the VRS movies
mvn NAME: Prints the VRS movies containing the name NAME
mvp PRICE: Prints the VRS movies with the renting price equal to PRICE
mvpge PRICE: Prints the VRS movies with the renting price greater than or equal to PRICE
mvpgt PRICE: Prints the VRS movies with the renting price greater than PRICE
mvple PRICE: Prints the VRS movies with the renting price less than or equal to PRICE
mvplt PRICE: Prints the VRS movies with the renting price less than PRICE
mvy YEAR: Prints the VRS movies with the release year equal to YEAR
mvyge YEAR: Prints the VRS movies with the release year greater than or equal to YEAR
mvygt YEAR: Prints the VRS movies with the release year greater than YEAR
mvyle YEAR: Prints the VRS movies with the release year less than or equal to YEAR
mvylt YEAR: Prints the VRS movies with the release year less than YEAR
version: Prints the VRSP version
quit: Quits VRSP
VRSP>
```

La commande `version` permet d'obtenir cet affichage :

```
$ ./vrsp.out vrs.xml
VRSP> version
VRSP (Video Rental Shop Program) 20181127

Copyright (C) 2018 Khaled Khebbeb and Samson Pierre.

```

```
7  Written by Khaled Khebbeb <khaled.khebbeb@univ-pau.fr> and Samson Pierre
   <samson.pierre@univ-pau.fr>.
8  VRSP>
```

Les prénoms `Khaled` et `Samson`, les noms `Khebbeb` et `Pierre` ainsi que les adresses e-mail `khaled.khebbeb@univ-pau.fr` et `samson.pierre@univ-pau.fr` sont à remplacer par les vôtres. La version `20181127` est à remplacer par la version de votre programme.

La commande `quit` permet de quitter le programme :

```
1  $ ./vrsp.out vrs.xml
2  VRSP> quit
3  $
```

Les autres commandes affichent un contenu spécifique au vidéo-club et aux films qu'il loue :

```
1  $ ./vrsp.out vrs.xml
2  VRSP> addr
3  Rent a video, 1 rue Richelieu, 64000, Pau
4  VRSP> mv
5  Alien (1979), 1.99 EUR
6  Aliens (1986), 1.99 EUR
7  Alien 3 (1992), 1.99 EUR
8  Alien Resurrection (1997), 1.99 EUR
9  Prometheus (2012), 3.99 EUR
10 Alien: Covenant (2017), 3.99 EUR
11 The Lord of the Rings: The Fellowship of the Ring (2001), 2.99 EUR
12 The Lord of the Rings: The Two Towers (2002), 2.99 EUR
13 The Lord of the Rings: The Return of the King (2003), 2.99 EUR
14 VRSP> mvn Alien
15 Alien (1979), 1.99 EUR
16 Aliens (1986), 1.99 EUR
17 Alien 3 (1992), 1.99 EUR
18 Alien Resurrection (1997), 1.99 EUR
19 Alien: Covenant (2017), 3.99 EUR
20 VRSP> mvp 2.99
21 The Lord of the Rings: The Fellowship of the Ring (2001), 2.99 EUR
22 The Lord of the Rings: The Two Towers (2002), 2.99 EUR
23 The Lord of the Rings: The Return of the King (2003), 2.99 EUR
24 VRSP> mvpge 2.99
25 Prometheus (2012), 3.99 EUR
26 Alien: Covenant (2017), 3.99 EUR
27 The Lord of the Rings: The Fellowship of the Ring (2001), 2.99 EUR
28 The Lord of the Rings: The Two Towers (2002), 2.99 EUR
29 The Lord of the Rings: The Return of the King (2003), 2.99 EUR
30 VRSP> mvpgt 2.99
31 Prometheus (2012), 3.99 EUR
32 Alien: Covenant (2017), 3.99 EUR
33 VRSP> mvple 2.99
34 Alien (1979), 1.99 EUR
35 Aliens (1986), 1.99 EUR
36 Alien 3 (1992), 1.99 EUR
37 Alien Resurrection (1997), 1.99 EUR
38 The Lord of the Rings: The Fellowship of the Ring (2001), 2.99 EUR
39 The Lord of the Rings: The Two Towers (2002), 2.99 EUR
40 The Lord of the Rings: The Return of the King (2003), 2.99 EUR
41 VRSP> mvplt 2.99
42 Alien (1979), 1.99 EUR
43 Aliens (1986), 1.99 EUR
44 Alien 3 (1992), 1.99 EUR
45 Alien Resurrection (1997), 1.99 EUR
46 VRSP> mvy 1997
47 Alien Resurrection (1997), 1.99 EUR
48 VRSP> mvyge 1997
49 Alien Resurrection (1997), 1.99 EUR
50 Prometheus (2012), 3.99 EUR
51 Alien: Covenant (2017), 3.99 EUR
52 The Lord of the Rings: The Fellowship of the Ring (2001), 2.99 EUR
53 The Lord of the Rings: The Two Towers (2002), 2.99 EUR
54 The Lord of the Rings: The Return of the King (2003), 2.99 EUR
55 VRSP> mvygt 1997
56 Prometheus (2012), 3.99 EUR
57 Alien: Covenant (2017), 3.99 EUR
58 The Lord of the Rings: The Fellowship of the Ring (2001), 2.99 EUR
59 The Lord of the Rings: The Two Towers (2002), 2.99 EUR
60 The Lord of the Rings: The Return of the King (2003), 2.99 EUR
61 VRSP> mvyle 1997
62 Alien (1979), 1.99 EUR
```

```
Aliens (1986), 1.99 EUR
Alien 3 (1992), 1.99 EUR
Alien Resurrection (1997), 1.99 EUR
VRSP> mvylt 1997
Alien (1979), 1.99 EUR
Aliens (1986), 1.99 EUR
Alien 3 (1992), 1.99 EUR
VRSP>
```

Si la commande saisie n'existe pas, le programme doit afficher un message d'erreur. Idem si le paramètre de la commande manque, si le paramètre de la commande est incorrect ou si la commande dépasse 18 caractères :

```
$ ./vrsp.out vrs.xml
VRSP> hello
./vrsp.out: Invalid command
VRSP> mvp
./vrsp.out: Missing parameter for the mvp command
VRSP> mvp abc
./vrsp.out: Invalid parameter for the mvp command
VRSP> mvp 012345678901234
./vrsp.out: Too many characters for the command
VRSP>
```

Les fichiers d'en-tête

Voici le fichier d'en-tête `movie.h` :

```c
/**
 * \file movie.h
 */
#ifndef MOVIE_H
#define MOVIE_H
/**
 * A movie.
 */
typedef struct
{
        char *name; /**< The movie name. */
        float price; /**< The movie renting price. */
        int year; /**< The movie release year. */
} movie_t;
/**
 * Creates a movie.
 * \return NULL on error (i.e., if the memory allocation is a failure), else a movie.
 */
movie_t *movie_create();
/**
 * Frees a movie.
 * \param movie The movie.
 */
void movie_free(movie_t *movie);
/**
 * Handles the mv command for a movie.
 * \param movie The movie.
 */
void movie_handle_mv(movie_t movie);
/**
 * Handles the mvn command for a movie.
 * \param movie The movie.
 * \param name The movie name.
 */
void movie_handle_mvn(movie_t movie, const char *name);
/**
 * Handles the mvp command for a movie.
 * \param movie The movie.
 * \param price The movie renting price.
 */
void movie_handle_mvp(movie_t movie, float price);
/**
 * Handles the mvpge command for a movie.
 * \param movie The movie.
 * \param price The movie renting price.
 */
void movie_handle_mvpge(movie_t movie, float price);
/**
```

```
 * Handles the mvpgt command for a movie.
 * \param movie The movie.
 * \param price The movie renting price.
 */
void movie_handle_mvpgt(movie_t movie, float price);
/**
 * Handles the mvple command for a movie.
 * \param movie The movie.
 * \param price The movie renting price.
 */
void movie_handle_mvple(movie_t movie, float price);
/**
 * Handles the mvplt command for a movie.
 * \param movie The movie.
 * \param price The movie renting price.
 */
void movie_handle_mvplt(movie_t movie, float price);
/**
 * Handles the mvy command for a movie.
 * \param movie The movie.
 * \param year The movie release year.
 */
void movie_handle_mvy(movie_t movie, int year);
/**
 * Handles the mvyge command for a movie.
 * \param movie The movie.
 * \param year The movie release year.
 */
void movie_handle_mvyge(movie_t movie, int year);
/**
 * Handles the mvygt command for a movie.
 * \param movie The movie.
 * \param year The movie release year.
 */
void movie_handle_mvygt(movie_t movie, int year);
/**
 * Handles the mvyle command for a movie.
 * \param movie The movie.
 * \param year The movie release year.
 */
void movie_handle_mvyle(movie_t movie, int year);
/**
 * Handles the mvylt command for a movie.
 * \param movie The movie.
 * \param year The movie release year.
 */
void movie_handle_mvylt(movie_t movie, int year);
#endif
```

Voici le fichier d'en-tête `vrs.h` :

```
/**
 * \file vrs.h
 */
#ifndef VRS_H
#define VRS_H
#include "movie.h" /* for movie_t */
/**
 * A VRS (Video Rental Shop).
 */
typedef struct
{
        char *city; /**< The VRS city. */
        movie_t **movies; /**< The VRS movies. */
        char *name; /**< The VRS name. */
        int nmovies; /**< The VRS number of movies. */
        int postal_code; /**< The VRS postal code. */
        char *street; /**< The VRS street. */
} vrs_t;
/**
 * Add a movie to a VRS.
 * \param vrs The VRS.
 * \param movie The movie.
 * \return -1 on error (i.e., if the memory allocation is a failure), else 0.
 */
int vrs_add_movie(vrs_t *vrs, movie_t *movie);
/**
```

```
27   * Creates a VRS.
28   * \return NULL on error (i.e., if the memory allocation is a failure), else a VRS.
29   */
30  vrs_t *vrs_create();
31  /**
32   * Frees a VRS.
33   * \param vrs The VRS.
34   */
35  void vrs_free(vrs_t *vrs);
36  /**
37   * Handles the addr command for a VRS.
38   * \param vrs The VRS.
39   */
40  void vrs_handle_addr(vrs_t vrs);
41  /**
42   * Handles the mv command for all the movies of a VRS.
43   * \param vrs The VRS.
44   */
45  void vrs_handle_mv(vrs_t vrs);
46  /**
47   * Handles the mvn command for all the movies of a VRS.
48   * \param vrs The VRS.
49   * \param name The VRS name.
50   */
51  void vrs_handle_mvn(vrs_t vrs, const char *name);
52  /**
53   * Handles the mvp command for all the movies of a VRS.
54   * \param vrs The VRS.
55   * \param price The movie renting price.
56   */
57  void vrs_handle_mvp(vrs_t vrs, float price);
58  /**
59   * Handles the mvpge command for all the movies of a VRS.
60   * \param vrs The VRS.
61   * \param price The movie renting price.
62   */
63  void vrs_handle_mvpge(vrs_t vrs, float price);
64  /**
65   * Handles the mvpgt command for all the movies of a VRS.
66   * \param vrs The VRS.
67   * \param price The movie renting price.
68   */
69  void vrs_handle_mvpgt(vrs_t vrs, float price);
70  /**
71   * Handles the mvple command for all the movies of a VRS.
72   * \param vrs The VRS.
73   * \param price The movie renting price.
74   */
75  void vrs_handle_mvple(vrs_t vrs, float price);
76  /**
77   * Handles the mvplt command for all the movies of a VRS.
78   * \param vrs The VRS.
79   * \param price The movie renting price.
80   */
81  void vrs_handle_mvplt(vrs_t vrs, float price);
82  /**
83   * Handles the mvy command for all the movies of a VRS.
84   * \param vrs The VRS.
85   * \param year The movie release year.
86   */
87  void vrs_handle_mvy(vrs_t vrs, int year);
88  /**
89   * Handles the mvyge command for all the movies of a VRS.
90   * \param vrs The VRS.
91   * \param year The movie release year.
92   */
93  void vrs_handle_mvyge(vrs_t vrs, int year);
94  /**
95   * Handles the mvygt command for all the movies of a VRS.
96   * \param vrs The VRS.
97   * \param year The movie release year.
98   */
99  void vrs_handle_mvygt(vrs_t vrs, int year);
100 /**
101  * Handles the mvyle command for all the movies of a VRS.
102  * \param vrs The VRS.
```

```
 * \param year The movie release year.
 */
void vrs_handle_mvyle(vrs_t vrs, int year);
/**
 * Handles the mvylt command for all the movies of a VRS.
 * \param vrs The VRS.
 * \param year The movie release year.
 */
void vrs_handle_mvylt(vrs_t vrs, int year);
#endif
```

Le fichier XML

Voici le fichier XML `vrs.xml` :

```
<vrs name="Rent a video">
        <street>1 rue Richelieu</street>
        <postal-code>64000</postal-code>
        <city>Pau</city>
        <movies>
                <movie name="Alien">
                        <year>1979</year>
                        <price>1.99</price>
                </movie>
                <movie name="Aliens">
                        <year>1986</year>
                        <price>1.99</price>
                </movie>
                <movie name="Alien 3">
                        <year>1992</year>
                        <price>1.99</price>
                </movie>
                <movie name="Alien Resurrection">
                        <year>1997</year>
                        <price>1.99</price>
                </movie>
                <movie name="Prometheus">
                        <year>2012</year>
                        <price>3.99</price>
                </movie>
                <movie name="Alien: Covenant">
                        <year>2017</year>
                        <price>3.99</price>
                </movie>
                <movie name="The Lord of the Rings: The Fellowship of the Ring">
                        <year>2001</year>
                        <price>2.99</price>
                </movie>
                <movie name="The Lord of the Rings: The Two Towers">
                        <year>2002</year>
                        <price>2.99</price>
                </movie>
                <movie name="The Lord of the Rings: The Return of the King">
                        <year>2003</year>
                        <price>2.99</price>
                </movie>
        </movies>
</vrs>
```