/\*

\* Brief: Header file with shared information

\* Autors: Catarina Ramos, Mário Fernandes;

\* Class: 5

\* Group: 14

\*/

#ifndef SOPE\_PROJ2\_RESOURCES\_H

#define SOPE\_PROJ2\_RESOURCES\_H

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <errno.h>

#include <sys/types.h>

#include <dirent.h>

#include <unistd.h>

#include <fcntl.h>

#include <sys/stat.h>

#include <limits.h>

#include <time.h>

#include <sys/wait.h>

#include <pthread.h>

#include <semaphore.h>

/\*

\* GENERIC ERROR MESSAGE

\*/

#define error(message) \

do { \

perror(message); \

exit(EXIT\_FAILURE);\

} while (0)

/\*

\* CAR INFO (GERADOR.LOG)

\*/

struct generator\_info{

clock\_t time;

int id\_car;

char destination;

clock\_t parking\_time;

clock\_t duration;

char state[16]; //e - enter / entrada , x - exit / saida , f - full / cheio , c - closed / encerrado

};

/\*

\* PARK STATUS (PARQUE.LOG)

\*/

struct park\_info{

clock\_t time;

int space;

int id\_car;

char \* state; //entrada , saida , cheio , encerrado

};

/\*

\* THREAD TIMER

\* Recieves the argument in clock\_t and makes the count down.

\*/

void \* timer(void \* argc){

clock\_t start\_time, current\_time;

clock\_t end\_time = \*(clock\_t \*) argc;

if ((start\_time = clock()) == -1)

error("Couldn't start clock\n");

if ((current\_time = clock()) == -1)

error("Couldn't start clock\n");

current\_time -= start\_time;

while(current\_time < end\_time) {

if ((current\_time = clock()) == -1)

error("Couldn't start clock\n");

current\_time -= start\_time;

}

return NULL;

};

#endif //SOPE\_PROJ2\_RESOURCES\_H