```
//
//  TLS_Lib
//      Explicit Models for Thread-Level Speculation
//
//  File    : TLS_Lib.h
//  Author  : Gabriel Campero
//          : Universidad de Los Andes, Ing. de Sist. Merida- Venezuela.
//  Purpose : Collection of classes offering explicit uses of TLS (thread-.
//            level speculation) for the extraction of parallelism.
//
//  Classes included: two_branches_speculator, multibranch_speculator, loop_speculator,
//                    critical_section_speculator.
//
// Limitations:
// * If a speculative thread is to be canceled, it cannot use functions
// that involve system mutexes, such as printf, etc. In this case, it
// is possible that the thread can be canceled while holding such a mutex,
// and the application con go into deadlock. In order to prevent this the
// user has to surround this "dangerous" code with:
//      "pthread_setcancelstate(PTHREAD_CANCEL_DISABLE, NULL);" and
//     "pthread_setcancelstate(PTHREAD_CANCEL_ENABLE, NULL);".
//
// * Sequential consistency is mostly guaranteed, save for exception
// behaviour.
//

#include <vector>
// Definition of type script_function, that masks a void* (*)(void*)
typedef void* (*script_function)(void*);
// Definition of type script_vector, a std::vector of type script_function
typedef std::vector<script_function> script_vector;



#include "_single_data_readers_log.h"
#include "_data_iterative_access_log.h"

#include "_previousSection.h"
#include "_speculativeBranch.h"
#include "_speculativeIteration.h"

#include "_conditional_speculator.h"
#include "two_branches_speculator.h"
#include "multibranch_speculator.h"
#include "loop_speculator.h"

#include "critical_section_speculator.h"
```