Help

```
#ifndef _WRAPPER_HPP
#define _WRAPPER_HPP

#include "pnl/pnl_random.h"
#include "
href../../../../common/math/jlparser/include/jlparser/parser_h_src.pdfjlparser/p

/**
 * Run the right MonteCarlo routine
 *
 * @param rng PnlRng
 * @param map an instance of Param
 * @param delta a boolean to decide whether to compute the delta
 * @param sample_average a boolean to decide whether to use IS with sample
 * averaging.
 * @param price a boolean to tell to only compute the price
 * @param couple a boolean to tell to couple RM and MC
 * @param average a boolean to tell to use an averaging RM method
 * @param poisson_only a boolean to decide to only apply the importance
 * sampling to the Poisson part
 * @param poisson a boolean to decide to apply the importance
 * sampling both to the Poisson part and the Brownian part
 * @param reduced a boolean telling to use a reduced importance sampling
 * parameter
 */
void MonteCarloWrapper(PnlRng *rng, const Param &map, bool price, bool delta,
                       bool sample_average, bool couple, bool average, bool pois
                       bool poisson, double &prix, double &std_dev);

void MultiLevelMonteCarloWrapper(PnlRng *rng, const Param &map, bool use_IS, dou

extern "C"{

#include "
href../../../../common/optype_h_src.pdfoptype.h"
PnlVect* ComputeEuropeanBSDrift(PnlVect *spot, PnlVect *sig, PnlVect *divid, dou
}

#endif /* _WRAPPER_HPP */
```