

Help

```

#if defined(PremiaCurrentVersion) && PremiaCurrentVersion <
    (2008+2) //The "#else" part of the code will be freely av
    ailable after the (year of creation of this file + 2)
#else
/*****
*   CPS - A simple C PDE solver                                *
*                                                                *
*   Copyright (c) 2007,                                        *
*   Maya Briani      <m.briani@iac.rm.cnr.it>,                *
*                                                                *
*   Francesco Ferreri <francesco.ferreri@gmail.com>,          *
*   Roberto Natalini <r.natalini@iac.rm.cnr.it>,              *
*   Marco Papi       <m.papi@iac.rm.cnr.it>                   *
*                                                                *
*****/
#ifndef PDE_PROBLEM_H
#define PDE_PROBLEM_H

#include "cps_types.h"
#include "cps_dimensions.h"

#define MAX_FILENAME 32

struct pde_problem_t {

    double desired_accuracy;
    unsigned int max_explicit_steps;
    unsigned int solution_size;

    boundary_description *boundary;
    pde *equation;
    grid *discretization_grid;
    problem_solver *solver;
    /* status access */
    int plotting_enabled;
    char plotfile[MAX_FILENAME];
};

int pde_problem_create(pde_problem**);

```

```
int pde_problem_destroy(pde_problem**);
int pde_problem_setup(pde_problem*);
int pde_problem_set_desired_accuracy(pde_problem *, double)
;
int pde_problem_set_equation(pde_problem *, pde *);
int pde_problem_set_grid(pde_problem *, grid *);
int pde_problem_set_boundary(pde_problem *, boundary_description *);
int pde_problem_solve(pde_problem*);
int pde_problem_get_solution(pde_problem *, double*);
int pde_problem_get_delta_x(pde_problem *, double*);
int pde_problem_set_plotting(pde_problem *, int);
int pde_problem_set_plotfile(pde_problem *, const char *);
int pde_problem_plot_solution(const pde_problem *);
#endif

#endif //PremiaCurrentVersion
```

References