

Help

```

#if defined(PremiaCurrentVersion) && PremiaCurrentVersion < (2008+2) //The "#els
#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>                  *
*                                                           *
*****/
#include "cps_pde_term.h"
#include "cps_stencil.h"
#include "cps_stencil_operator.h"
#include "cps_utils.h"
#include "cps_assertions.h"

int pde_term_create(pde_term **term, int type, function *f, stencil_operator *s)
{

    REQUIRE("function_not_null", f != NULL);
    REQUIRE("stencil_operator_not_null", s != NULL);

    STANDARD_CREATE(term, pde_term);
    (*term)->type = type;
    (*term)->function_factor = f;
    (*term)->st_operator = s;

    return OK;
}

int pde_term_destroy(pde_term **term)
{
    /* destroy term and associated stencil objects */

    if ((*term)->generated_stencil)
        stencil_destroy(&((*term)->generated_stencil));
    if ((*term)->st_operator)

```

```
    stencil_operator_destroy(&((*term)->st_operator));
    STANDARD_DESTROY(term);

    return OK;
}

int pde_term_set_function_factor(pde_term *pterm, const function *factor)
{
    /* set function factor */
    REQUIRE("pde_term_not_null", (pterm != NULL));
    REQUIRE("factor_not_null", (factor != NULL));

    pterm->function_factor = factor;

    return OK;
}

int pde_term_set_stencil_operator(pde_term *pterm, stencil_operator *stnop)
{
    /* set stencil operator for term */
    REQUIRE("pde_term_not_null", (pterm != NULL));
    REQUIRE("stencil_operator_not_null", (stnop != NULL));

    pterm->st_operator = stnop;

    return OK;
}

int pde_term_create_stencil(pde_term *pterm, const grid *g)
{
    /* apply stencil operator to create stencil for given term */
    REQUIRE("pde_term_not_null", (pterm != NULL));

    stencil_operator_apply(pterm->st_operator, pterm, g);
    pterm->generated_stencil = pterm->st_operator->applied_stencil;

    ENSURE("stencil_term_created", (pterm->generated_stencil != NULL));
    return OK;
}

/* end -- pde_term.c */
```

```
#endif //PremiaCurrentVersion
```