

Definície typov a konštánt pre jazyk C

```
//*****
//                                     (C) IpeSoft s.r.o. (Ltd.) ZILINA
// PROJECT : D2000
// FILE : Imp_def.h
//
// DESCRIPTION : definicia konstant a typov pre import gr.formatov
//
//*****

//*****
// Constants of gr.object's types
//*****
#define cLine = 0 // objekt ciara
#define cPLine = 1 // objekt viacnasobna ciara
#define cDLine = 2 // objekt viacnasobna prerusovana ciara
#define cArc = 3 // objekt obluk
#define c3Arc = 4 // objekt trojbodovy obluk
#define cBox = 5 // objekt obdlznik
#define cPAngle = 6 // objekt mnohouholnik
#define cCircle = 7 // objekt kruh
#define cPiArc = 8 // objekt kruhova vysec
#define cEllipse = 11 // objekt elipsa
#define cText = 12 // objekt text
#define cGroup = 32 // skupina objektov

// extra types
#define cLineCombined = -1 // skombinuje seriu ciar rovnakych atributov do jedneho objektu
// typu PolyLine alebo PolyLineDisjoint
#define cPLineAdd = -2 // pridať seriu parametrov do vytvaraneho objektu typu polyline
// (ak nie je vytvarany, tak založ nový)

#define cFontStyle = -3 // vytvorenie textového štýlu + autovytváranie
// poas vytvárania textov - od ver. 5.00
// (s daným menom môže existovať len 1 štýl)

//*****
// konstanty definujúce povolené typy pre niektoré parametre
//-----
// oznacenia v komentároch
// (*) - default parameter
//*****

//*****
// color definition, implicitne farby logickej palety pouzivanej pri importe
//*****

// bazy farieb
#define CLR_BASE_WHITE = 0
#define CLR_BASE_YELLOW = 16
#define CLR_BASE_CYAN = 32
#define CLR_BASE_GREEN = 48
#define CLR_BASE_RED = 64
#define CLR_BASE_PINK = 80
#define CLR_BASE_BLUE = 96

// bazy farieb - skratky
#define W = CLR_BASE_WHITE
#define Y = CLR_BASE_YELLOW
#define C = CLR_BASE_CYAN
#define G = CLR_BASE_GREEN
#define R = CLR_BASE_RED
#define P = CLR_BASE_PINK
#define B = CLR_BASE_BLUE

type TColorArr is ARRAY (0.. 15) OF integer // svetla..tmava
// indexy farieb
CLR_WHITE : constant TColorArr := (W+0,W+1,W+2,W+3,W+4,W+5,6,W+7,W+8,W+9,W+10,W+11,W+12,W+13,W+14,W+15)
CLR_YELLOW : constant TColorArr := (Y+0,Y+1,Y+2,Y+3,Y+4,Y+5,6,Y+7,Y+8,Y+9,Y+10,Y+11,Y+12,Y+13,Y+14,Y+15)
CLR_CYAN : constant TColorArr := (C+0,C+1,C+2,C+3,C+4,C+5,6,C+7,C+8,C+9,C+10,C+11,C+12,C+13,C+14,C+15)
CLR_GREEN : constant TColorArr := (G+0,G+1,G+2,G+3,G+4,G+5,6,G+7,G+8,G+9,G+10,G+11,G+12,G+13,G+14,G+15)
CLR_RED : constant TColorArr := (W+0,W+1,W+2,W+3,W+4,W+5,6,W+7,W+8,W+9,W+10,W+11,W+12,W+13,W+14,W+15)
CLR_PINK : constant TColorArr := (P+0,P+1,P+2,P+3,P+4,P+5,6,P+7,P+8,P+9,P+10,P+11,P+12,P+13,P+14,P+15)
CLR_BLUE : constant TColorArr := (B+0,B+1,B+2,B+3,B+4,B+5,6,B+7,B+8,B+9,B+10,B+11,B+12,B+13,B+14,B+15)
```

```

// niektore vybrane farby
#define i_CLR_WHITE = 0
#define i_CLR_PALEGRAYLIGHT = 2
#define i_CLR_PALEGRAY = 4
#define i_CLR_DARKGRAY = 10
#define i_CLR_DARKGRAYDARK = 12
#define i_CLR_BLACK = 15

#define i_CLR_YELLOW = 22
#define i_CLR_DARKYELLOW = 26
#define i_CLR_CYAN = 38
#define i_CLR_DARKCYAN = 42
#define i_CLR_GREEN = 54
#define i_CLR_DARKGREEN = 58
#define i_CLR_RED = 70
#define i_CLR_DARKRED = 74
#define i_CLR_PINK = 86
#define i_CLR_DARKPINK = 90
#define i_CLR_BLUE = 102
#define i_CLR_DARKBLUE = 106

//*****
// štýl iar
// lineStyle - type TPenStyle is (Solid, Alternate, Dash, Dot, DashDot, DashDotDot, Invisible)
//*****
#define tLS_Solid = 0 // ----- (*)
#define tLS_Alternate = 1 // . . . . . - iba tenka, nepouzivat
#define tLS_Dash = 2 // - - - - -
#define tLS_Dot = 3 // . . . . .
#define tLS_DashDot = 4 // - . - . -
#define tLS_DashDotDot = 5 // - . . - .
#define tLS_Invisible = 6 // nepodporena

//*****
// ukonenie iar
// lineEnd - type TLineEnd is (Flat, Square, Round)
//*****
#define tLE_Flat = 0 // ostry roh (*)
#define tLE_Square = 1 // skoseny roh
#define tLE_Round = 2 // zaobleny roh

//*****
// zalomenie iar
// lineJoin - type TLineJoin is (Bevel, Round, Miter)
//*****
#define tLJ_Bevel = 0 // ukoncenie na koncovom bode (*)
#define tLJ_Round = 1 // zaoblene zakoncenie za koncovym bodom
#define tLJ_Miter = 2 // hranate zakoncenie za koncovym bodom

//*****
// vyplnove vzory poskytovane vo Windows
// brushStyle - type TBrushStyle is (Solid, Hollow, BDiagonal, Cross, DiagCross,
// FDiagonal, Horizontal, Vertical)
//*****
#define tBS_Solid = 0 // plny vzor
#define tBS_Hollow = 1 // prazdny vzor (*)
#define tBS_BDiagonal = 2
#define tBS_Cross = 3
#define tBS_DiagCross = 4
#define tBS_FDiagonal = 5
#define tBS_CHorizontal = 6
#define tBS_Vertical = 7

//*****
// umiestnenie textov
// cTextPos - type tTextPos is (tpAtPos, tpInBox, tpIntoBox)
//*****
#define tTP_ATPOS = 0 // text na zadanej pozicii (*)
#define tTP_INBLOCK = 1 // text umiestneny v obdlniku
#define tTP_INTBLOCK = 2 // text zaplnujuci obdlnik

//*****
// horizontalne umiestnenia pre text umiestneny v obdlniku
// cTextCenterH -horizontal center
//*****
#define tTHC_LEFT = 0 // vlavo
#define tTHC_MIDDLE = 1 // centrovany (*)
#define tTHC_RIGHT = 2 // vpravo

```

```

//*****
// vertikálne umiestnenia pre text umiestneny v obdĺžniku
// cTextCenterV - vertical center
//*****
#define tTVC_TOP          = 0 // hore
#define tTVC_MIDDLE       = 1 // centrovany (*)
#define tTVC_BOTTOM      = 2 // dole

//*****
// parametre gr.objektov - param's types constants
//-----
// implementacne poznamky
// 1. aby boli parametre akceptovane, musia by v procedurach takych typov ako
//    su v poznamke za parametrom
// 2. nie vsetky parametre su realizovane
// 3. možno zada max.1000 pozicnych bodov, ale V4.5 pozna max.prvych 1+30
//    pri vašom pote bodov pre typ polyline sa generuje viacero objektov tohto typu
// 4. index farby pre import index do default farebnej palety
// 5. farba RGB od ver V5.0 farby plnia 128 prvkovu tabulku z ktorej bude vytvoreny
//    objekt paleta, farby v objekte budu ukladane ako indexy do tejto tabulky
//*****

//-----
// positions and sizes
//-----
#define cPosXY             =1 // 2 * int/float - pozicia bodu
#define cPosDX             =2 // 2 * int/float - vzdialenost bodu od predosleho bodu

//-----
// line params
//-----
#define cLineColorRGB      =10 // int - RGB farba objektu - od ver V5.0
#define cLineColorIdx      =11 // int - index farby v lokalnej palete
#define cLineStyle         =12 // int - styl ciary - lineStyles
#define cLineWidth        =13 // int - hrubka ciary 1..
#define cLineEnd           =14 // int - zakoncenie hrubej ciary - lineEnd
#define cLineJoin          =15 // int - zalomenie polyline ciary - lineJoin

//-----
// fill params
//-----
#define cFillColorRGB      =30 // int - RGB farba objektu - od ver V5.0
#define cFillColorIdx      =31 // int - index farby v lokalnej palete
#define cFillPattern       =32 // int - styl ciary - brushStyle

//-----
// circles
//-----
#define cCircleRadial      =50 // int/float - polomer kruhu

#define cCircleAngleDegStart =56 // int/float - degree
#define cCircleAngleDegEnd   =57 // int/float - degree
#define cCircleAngleDegSize  =58 // int/float - degree

#define cCircleAngleRadStart =59 // int/float - radiant
#define cCircleAngleRadEnd   =60 // int/float - radiant
#define cCircleAngleRadSize  =61 // int/float - degree

//-----
// text params
//-----
#define cTextText          =70 // text
#define cTextColorRGB      =71 // int - RGB farba objektu - od ver V5.0
#define cTextColorIdx      =72 // int - index farby v lokalnej palete

#define cTextPos           =73 // int - umiestnenie v obdĺžniku, tTP_ATPOS,..
#define cTextCenterH       =76 // int - horizontálne centrovanie v obdĺžniku
#define cTextCenterV       =77 // int - vertikálne centrovanie v obdĺžniku

```

```

//*****
// vytvorenie textového štýlu
// txt.štýl sa vytvorí ako objekt s parametrom CFont alebo pocas vytvarania textu
// * ak nebudú nastavené všetky parametre font sa nevytvorí
// * ak štýl s menom v cFontStyleName existuje, nový nebude vytvorený, použije sa existujúci
// * ak je txt.štýl vytvorený počas vytvárania textu, bude text kreslený týmto štýlom
// * ak je počas vytvárania textu zadaný iba cFontStyleName,
// potom bude text kreslený týmto štýlom (ak existuje)
//*****
#define cFontStyleName      =200 // string - meno textového štýlu
#define cFontName           =201 // string - meno fontu
#define cFontSize           =202 // Integer - vyska fonty
#define cFontBold           =203 // Boolean - atribut pre hruby font
#define cFontItalic         =204 // Boolean - atribut pre sikmy font
#define cFontUnderline      =205 // Boolean - atribut pre podtrhnuty font
#define cFontStrikeOut      =206 // Boolean - atribut pre preciar knuty font
#define cFontCharSet        =207 // Integer - subor znakov (Western,Central euroean)

//*****
// types of obj.actions
//*****
type tObjAction is (closeFigure,// uzatvor polyline , prevod na mnohouholník
                   closeObject,// zatvor objekt
                   closeGroup, // zatvor objekt a grupu
                   closeAll) // zatvor objekt a všetky grupy
                        // vnútorne volaný po ukonení importu

//*****
// access to functions - definitions
//*****

void _stdcall (*CreateObj)(int)
void _stdcall (*ObjAction)(tObjAction)

void _stdcall (*Set_string )(int,char far * )
void _stdcall (*Set_boolean)(int,bool)
void _stdcall (*Set_integer)(int,int)
void _stdcall (*Set_float)(int,real)
void _stdcall (*Set_integer2)(int,int,int)
void _stdcall (*Set_float2)(int,real,real)

// info text pocas importu
void _stdcall (*ShowInfo)(char far * )

//*****
// access to functions
//*****
createObj      : tCreateObj      := null;
objAction      : tObjAction      := null;

set_string     : tset_string     := null;
set_boolean    : tset_boolean    := null;
set_integer    : tset_integer    := null;
set_float      : tset_float      := null;
set_integer2   : tset_integer2   := null;
set_float2     : tset_float2     := null;

// info text pocas importu
ShowInfo       : tShowInfo       := null;

//*****
// access to import functions
//*****

#define pShowInfo      = 0
#define pcreateObj     = 1
#define pObjAction     = 2
#define pset_string    = 3
#define pset_boolean   = 4
#define pset_integer   = 5
#define pset_float     = 6
#define pset_integer2  = 7
#define pset_float2    = 8

// pointer to 1.char of null terminating string
#define maxResStr      = 10000

// nazov importu a tup suboru
void _stdcall (*GetFileType)(char far * description,char far * extension)

// inicializacia call-back procedur
void _stdcall (*ImportConnect)(int procType,int procAddr )

```

```
// load autocad dxf file
void _stdcall (*ImportConnect)(int procType,int procAddr )
```

```
//*****
//-----
// Revisions History --
//-----
// 0.00      28.02.01 - vytvorenie modulu
//-----
```



Súvisiace stránky:

[Import vektorových formátov do schém systému D2000](#)