

- (1) Pentru a porni/inițializa un sistem grafic se apelează mai întâi funcția **initgraph**. Aceasta inițializează sistemul grafic prin încărcarea driver-ului grafic de pe disc și pune sistemul în modul grafic. În mod normal, *initgraph* încarcă un driver grafic prin alocare de memorie pentru respectivul driver (prin **\_graphgetmem**) și încărcarea apoi, de pe disc, a **fișierului .BGI** adecvat. În **Tabel 1** și **Tabel 2** sunt enumerate driver-ele grafice, respectiv, modulele grafice și paletele de culori așa după cum apar ele definite în *graphics.h*:

**Tabel 1:** Driver-ele grafice

Constanta - driver grafic	Valoare numerică
DETECT	0 (solicită autodetecție)
CGA	1
MCGA	2
EGA	3
EGA64	4
EGAMONO	5
IBM8514	6
HERCMONO	7
ATT400	8
VGA	9
PC3270	10

**Tabel 2:** Modulele grafice și paletele de culori aferente

Driver grafic	Modul grafic	Val.	Coloane x Linii	Paleta	Pagini <sup>1</sup>
CGA	CGAC0	0	320 x 200	C0	1
	CGAC1	1	320 x 200	C1	1
	CGAC2	2	320 x 200	C2	1
	CGAC3	3	320 x 200	C3	1
	CGAHI	4	640 x 200	2 culori	1
MCGA	MCGAC0	0	320 x 200	C0	1
	MCGAC1	1	320 x 200	C1	1
	MCGAC2	2	320 x 200	C2	1
	MCGAC3	3	320 x 200	C3	1
	MCGAMED	4	640 x 200	2 culori	1
	MCGAHI	5	640 x 480	2 culori	1
EGA	EGALO	0	640 x 200	16 culori	4
	EGAHI	1	640 x 350	16 culori	2
EGA64	EGA64LO	0	640 x 200	16 culori	1
	EGA64HI	1	640 x 350	4 culori	1
EGA-MONO	EGAMONHI	3	640 x 350	2 culori	1
	EGAMONHI	3	640 x 350	2 culori	2
HERC	HERCMONHI	0	720 x 348	2 culori	2
ATT400	ATT400C0	0	320 x 200	C0	1
	ATT400C1	1	320 x 200	C1	1
	ATT400C2	2	320 x 200	C2	1
	ATT400C3	3	320 x 200	C3	1
	ATT400MED	4	640 x 200	2 culori	1
	ATT400HI	5	640 x 400	2 culori	1
VGA	VGALO	0	640 x 200	16 culori	2
	VGAMED	1	640 x 350	16 culori	2
	VGACHI	2	640 x 480	16 culori	1
PC3270	PC3270HI	0	720 x 350	2 culori	1
IBM8514	IBM8514HI	0	640 x 480	256 culori	1
	IBM8514LO	0	1024 x 768	256 culori	1

- (2) Următorul tabel ilustrează codurile de eroare returnate de funcția **graphresult**:

Code	Constant	Corresponding error message string
0	grOk	No error
-1	grNoInitGraph	(BGI) graphics not installed (use initgraph)

<sup>1</sup> Numai cardurile grafice EGA, VGA și Hercules suportă *pagini multiple*.

-2	grNotDetected	Nu s-a detectat nici un hardware grafic
-3	grFileNotFound	Device driver file not found
-4	grInvalidDriver	Invalid device driver file
-5	grNoLoadMem	Not enough memory to load driver
-6	grNoScanMem	Out of memory in scan fill (eroare la umplerea suprafeței)
-7	grNoFloodMem	Out of memory in flood fill
-8	grFontNotFound	Font file not found
-9	grNoFontMem	Not enough memory to load font
-10	grInvalidMode	Invalid graphics mode for selected driver
-11	grError	Graphics error
-12	grIOerror	Graphics I/O error
-13	grInvalidFont	Invalid font file
-14	grInvalidFontNum	Invalid font number
-15	grInvalidDeviceNum	Invalid device number
-18	grInvalidVersion	Invalid version number

(3) Structura **fillsettingstype** este definită în *graphics.h* astfel:

```
struct fillsettingstype
{
    int pattern;    // pattern de umplere curent
    int color;      // culoare de umplere curentă
};
```

(4) Există 11 stiluri predefinite de pattern-uri de umplere iar numele lor simbolice sunt oferite în cadrul enumerației **fill\_patterns** menționată în *graphics.h* și redată mai jos:

Nume	Valoare	Descriere
EMPTY_FILL	0	Fill with background color
SOLID_FILL	1	Solid fill
LINE_FILL	2	Fill with ---
LTSLASH_FILL	3	Fill with ///
SLASH_FILL	4	Fill with ///, thick lines
BKSLASH_FILL	5	Fill with \\, thick lines
LTBKSLASH_FILL	6	Fill with \\\
HATCH_FILL	7	Light hatch fill
XHATCH_FILL	8	Heavy crosshatch fill
INTERLEAVE_FILL	9	Interleaving line fill
WIDE_DOT_FILL	10	Widely spaced dot fill
CLOSE_DOT_FILL	11	Closely spaced dot fill
USER_FILL	12	User-defined fill pattern

(5) Structura **linesettingstype** este definită în *graphics.h* astfel:

```
struct linesettingstype
{
    int linestyle;
    unsigned upattern;
    int thickness;
};
```

unde,

- **linestyle** specifică stilul de linie folosit în desenare (de ex., linie solidă, punctată cu buline, centrată, punctată). Enumerarea *line\_styles*, definită în `graphics.h` oferă numele acestor operatori. Pe lângă cele 4 stiluri standard există și unul definit de utilizator în care șablonul este o dată de tip *int* (16 biți). Fiecare bit nenul din șablon reprezintă un pixel colorat cu culoarea curentă iar biții nuli reprezintă pixeli de culoarea fondului.
- | Nume         | Valoare | Descriere               |
|--------------|---------|-------------------------|
| SOLID_LINE   | 0       | Solid line              |
| DOTTED_LINE  | 1       | Dotted line             |
| CENTER_LINE  | 2       | Centered line           |
| DASHED_LINE  | 3       | Dashed line             |
| USERBIT_LINE | 4       | User-defined line style |
- **grosimea** specifică dacă lățimea liniilor ce se vor desena va fi una normală sau mai groasă.
- | Nume        | Valoare | Descriere       |
|-------------|---------|-----------------|
| NORM_WIDTH  | 1       | 1 pixel lățime  |
| THICK_WIDTH | 3       | 3 pixeli lățime |

Parametrul *linestyle* nu afectează arcurile de cerc, cercurile, elipsele sau sectoarele de elipsă. În cazul acestora doar valoarea parametrului *thickness* (grosimea liniei) este utilizată.

- (6) Valorile constante acceptate pentru variabila *kind* furnizată ca parametru în funcția *ismouseclick* sunt următoarele:

Constanta <i>kind</i>	Tipul de eveniment specificat
WM_MOUSEMOVE	mișcarea mouse-ului
WM_LBUTTONDOWNCLK	dublu click pe butonul stâng al mouse-ului
WM_LBUTTONDOWN	click pe butonul stâng al mouse-ului
WM_LBUTTONUP	eliberarea butonului stâng al mouse-ului
WM_MBUTTONDOWNCLK	dublu click pe butonul din mijloc al mouse-ului
WM_MBUTTONDOWN	click pe butonul din mijloc al mouse-ului
WM_MBUTTONUP	eliberarea butonului din mijloc al mouse-ului
WM_RBUTTONDOWNCLK	dublu click pe butonul drept al mouse-ului
WM_RBUTTONDOWN	click pe butonul drept al mouse-ului
WM_RBUTTONUP	eliberarea butonului drept al mouse-ului