TEMA 2 - Camp

Autor:

Nume: Politic Andrei - Bogdan
Anul I; Grupa 315CD
Universitatea Politehnica Bucuresti
Facultatea de Automatica si Calculatoare
Profil: CTI

CUPRINS

EXERCITIUL 1
Reprezentarea campurilor scalare

Fie functia de doua variabile, f : $[0,\pi] \times [0,2\pi] \rightarrow$ R, f(x, y) = $(\sin(x) + \sin(y))^2$. Vom reprezenta tridimensional graficul functiei pentru domeniul dat, si de asemenea curbele de nivel ale graficului in format bidimensional si tridimensional.

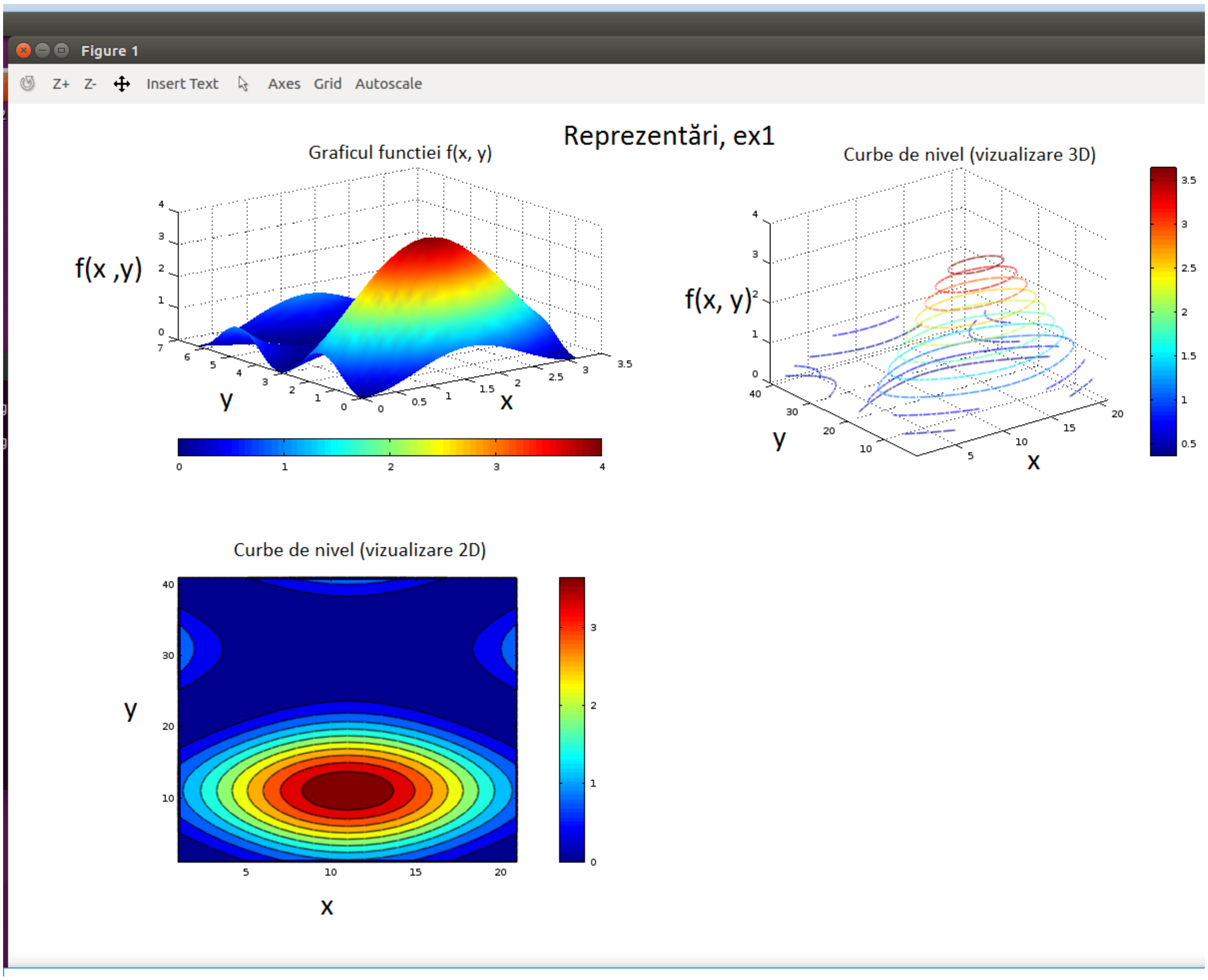

Figure 1: Graficul functiei si curbele de nivel (1)

Graficul, privit din alt unghi:

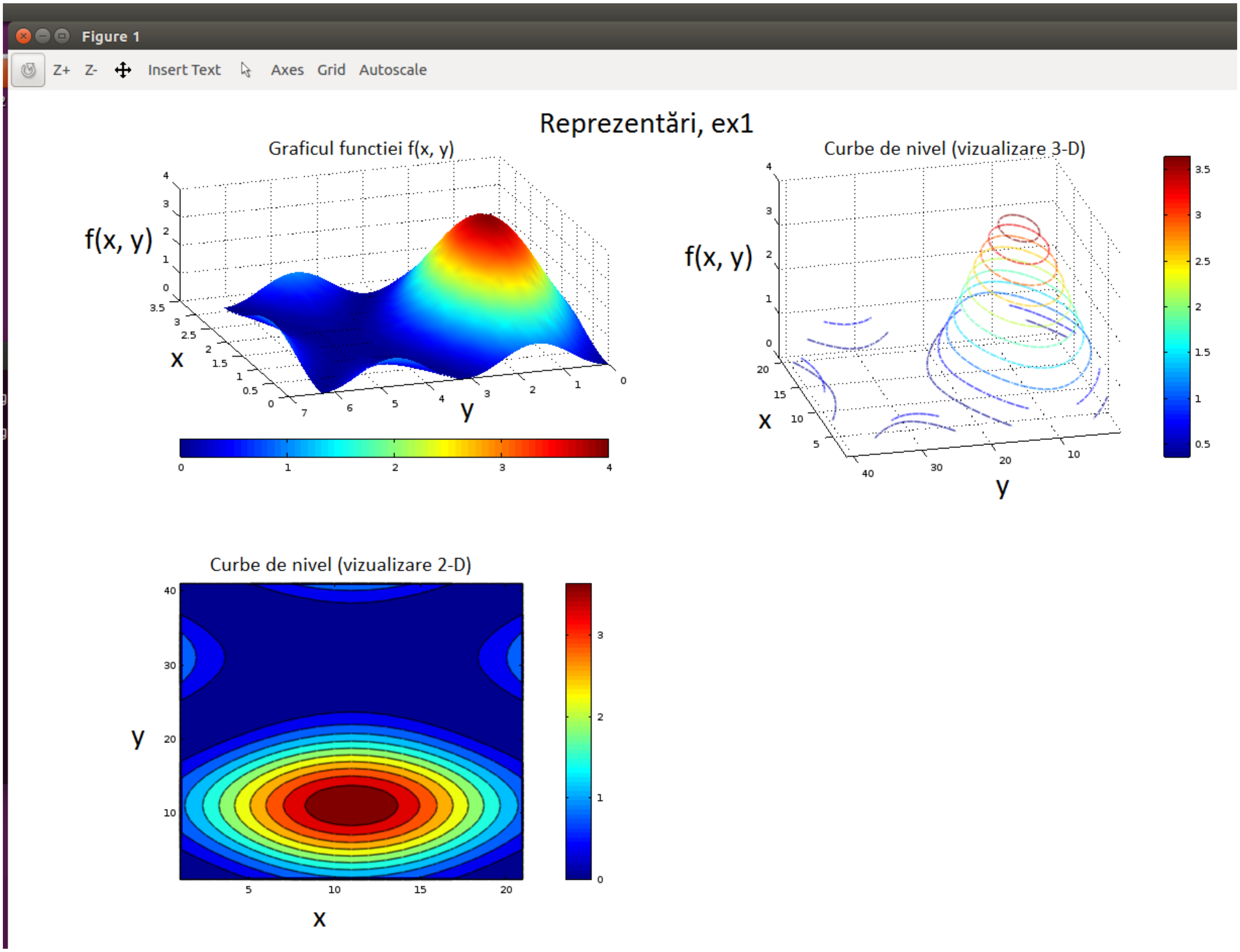

Figure 2: Graficul functiei si curbele de nivel (2)

EXERCITIUL 1 - COD OCTAVE

```
function scalar_field_representation ()
    x = linspace(0,pi,21);
y = linspace(0,2*pi,41);

    [X Y] = meshgrid(x, y);

    Z = (sin(X) + sin(Y)).^2;

    figure; hold;
colormap("default");
title(sprintf("Reprezentari, ex1"));

    subplot(2,2,1);
title(sprintf("Graficul functiei f(x,y) :"));
mesh(X,Y,Z);
xlabel(sprintf("x",'default'));
ylabel(sprintf("y"));
zlabel(sprintf("f(x, y)"));
shading interp;
colorbar("SouthOutSide");
subplot(2,2,2);
title(sprintf("Curbe de nivel (vizualizare 3-D) :"));
contour3(Z);
xlabel(sprintf("x"));
ylabel(sprintf("y"));
zlabel(sprintf("f(x, y)"));
colorbar("EastOutSide");
subplot(2,2,3);
title(sprintf("Curbe de nivel (vizualizare 2-D) :")); contourf(Z);
xlabel(sprintf("x"));
ylabel(sprintf("y"));
colorbar("EastOutSide");

    endfunction
```

EXERCITIUL 2

Reprezentarea campurilor vectoriale

Consideram functia vectoriala depinzand de doua variabile scalare, reale (x,y): G : [-10, 10] $\times$ [-10, 10] $\rightarrow R^2$, G(x,y) = G_x(x,y) $\cdot$ i + G_y(x,y) $\cdot$ j = 2xcos(y) $\cdot$ i + 3ysin(x) $\cdot$ j . Vom reprezenta spectrul (discret al) campului G, suprapus peste harta de valori a modulelor vectorilor ce alcatuiesc spectrul.

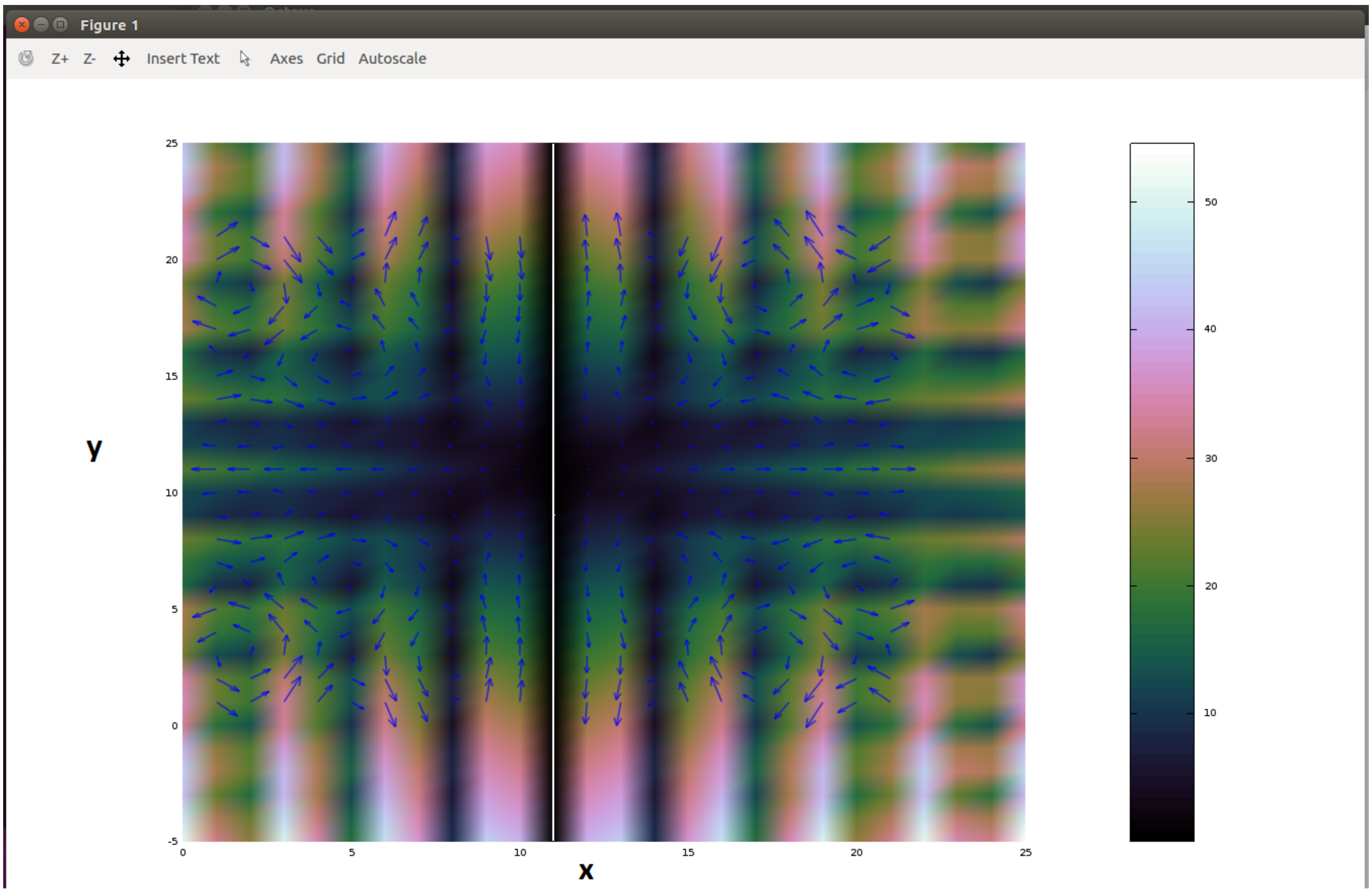

Figure 3: **Spectrul campului G (reprezentat cu albastru), suprapus peste harta de valori a modulului campului**

EXERCITIUL 2.a. - COD OCTAVE

```
function vectorial_field_representation ()
x = linspace(-10, 10, 21);
y = linspace(-10, 10, 21);
surfx = linspace(0, 25, 26);
surfy = linspace(-5, 25, 31);

[X Y] = meshgrid(x, y);
[surfX surfY] = meshgrid(surfx, surfy);
figure; hold; colormap(”cubehelix”);
Gmodule = sqrt( (2 * (surfX-11).* cos(surfY-11)).^2 + (3 * (surfY-11).* sin(surfX-11)).^2);
surf(surfX, surfY, Gmodule - 50);
shading interp;
G = quiver(2 * X.* cos(Y), 3 * Y.* sin(X));

colorbar(”EastOutSide”);

endfunction
```

In continuare, va fi prezentat graficul gradientului campului scalar f (f de la Exercitiul 1), suprapus peste reprezentarea curbelor de nivel (echivalorilor) graficului lui f.

. **Pe caz general, gradientul unei functii de variabila multipla este un vector ce contine derivatele partiale ale variabilei si care, in sens fizic, indica panta cu gradul cel mai mare de inclinare de pe graficul functiei, in sensul de urcare, intr-un punct anume. In cazul campului scalar reprezentat de valorile functiei f, gradientul apare sub forma vectorului [∂f / ∂x, ∂f / ∂y], adica este egal cu [∂($(sin(x) + sin(y))^2$) / ∂x, ∂($(sin(x) + sin(y))^2$) / ∂y]. Se obtine ca gradientul campului scalar f intr-un punct de coordonate (x, y) este [2 · (sin(x) + sin(y)) · cos(x), 2 · (sin(x) + sin(y)) · cos(y)] .**

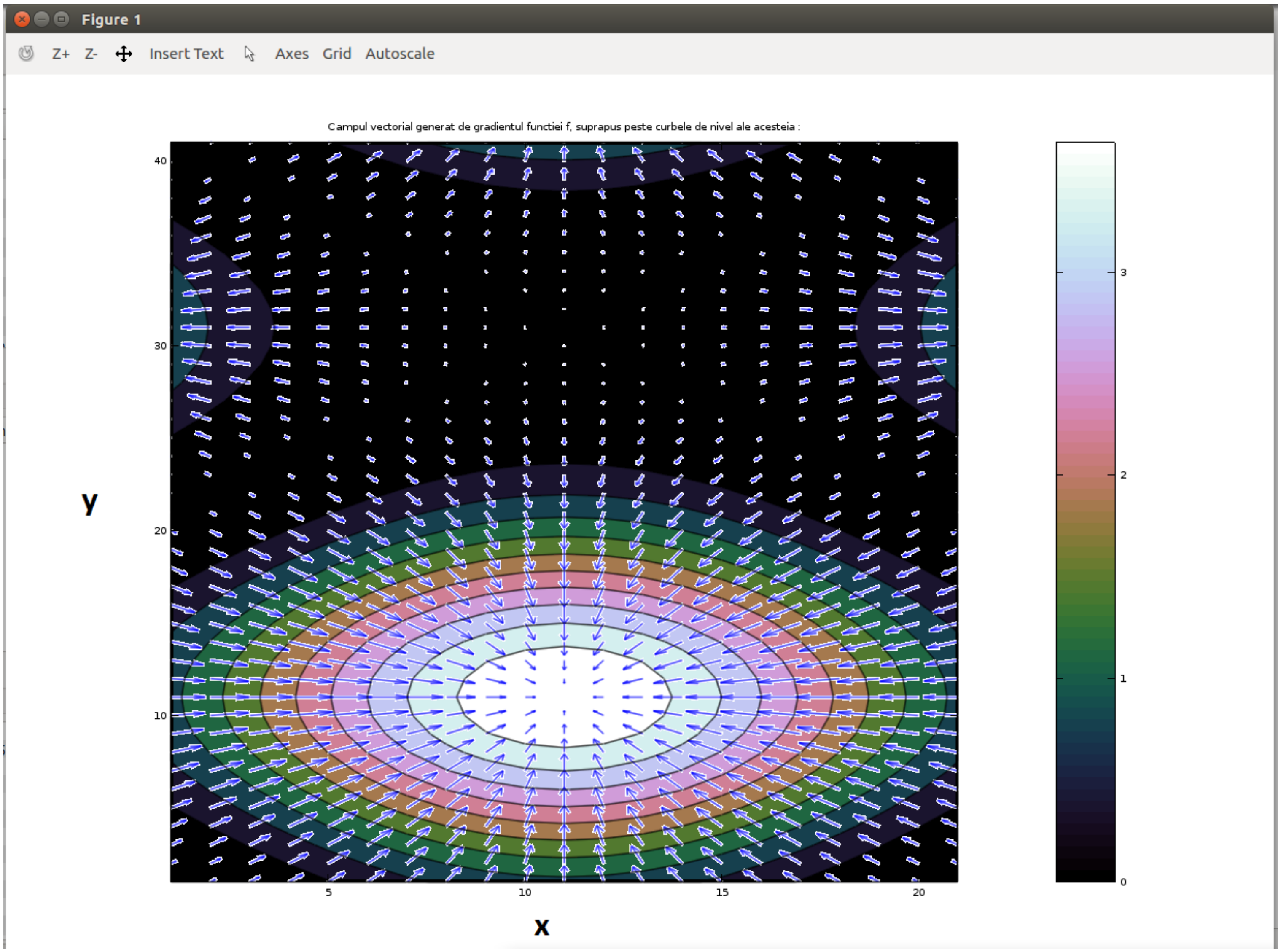

Figure 4: **Gradientul campului scalar f si echivalorile graficului lui f.**

EXERCITIUL 2.b. - COD OCTAVE

```
function gradient_f ()
x = linspace(0,pi,21);
y = linspace(0,2*pi,41);

[X Y] = meshgrid(x, y);

function_f = (sin(X) + sin(Y)).2̂;
derx = 2 * (sin(X) + sin(Y)).* cos(X);
dery = 2 * (sin(X) + sin(Y)).* cos(Y);

figure; hold;
colormap(”cubehelix”);

title(sprintf(”Campul vectorial generat de gradientul functiei f, suprapus peste curbele de nivel ale acesteia :”));
quiver(derx, dery);
contourf(function_f);
colorbar(”EastOutSide”);

endfunction
```

Vom calcula si vom reprezenta divergenta campului vectorial G. Pe caz general, divergenta reprezinta densitatea de flux dintr-o unitate infinit de mica de volum, specifica unui punct. Matematic, considerand un camp vectorial G, cu G(x,y) = G_x(x,y) $\cdot$ i + G_y(x,y) $\cdot$ j, divergenta acestuia este ∂G$_x$(x,y) / ∂x + ∂G$_x$(x,y) / ∂y .

. **In cazul nostru, divergenta lui G este** $2 \cdot cos(x) + 3 \cdot sin(y)$ **.**

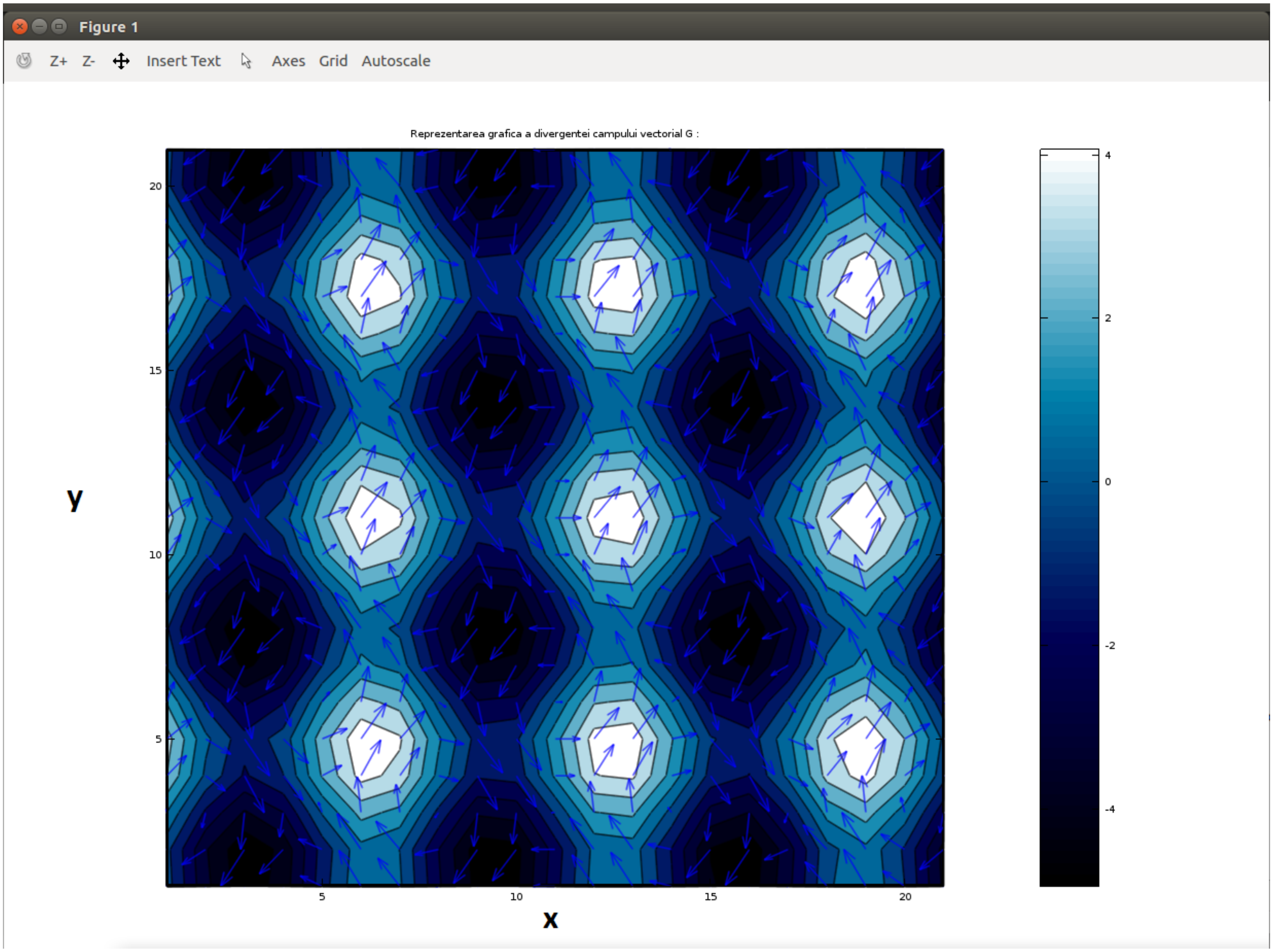

Figure 5: **Divergenta campului vectorial G (bidimensional).**

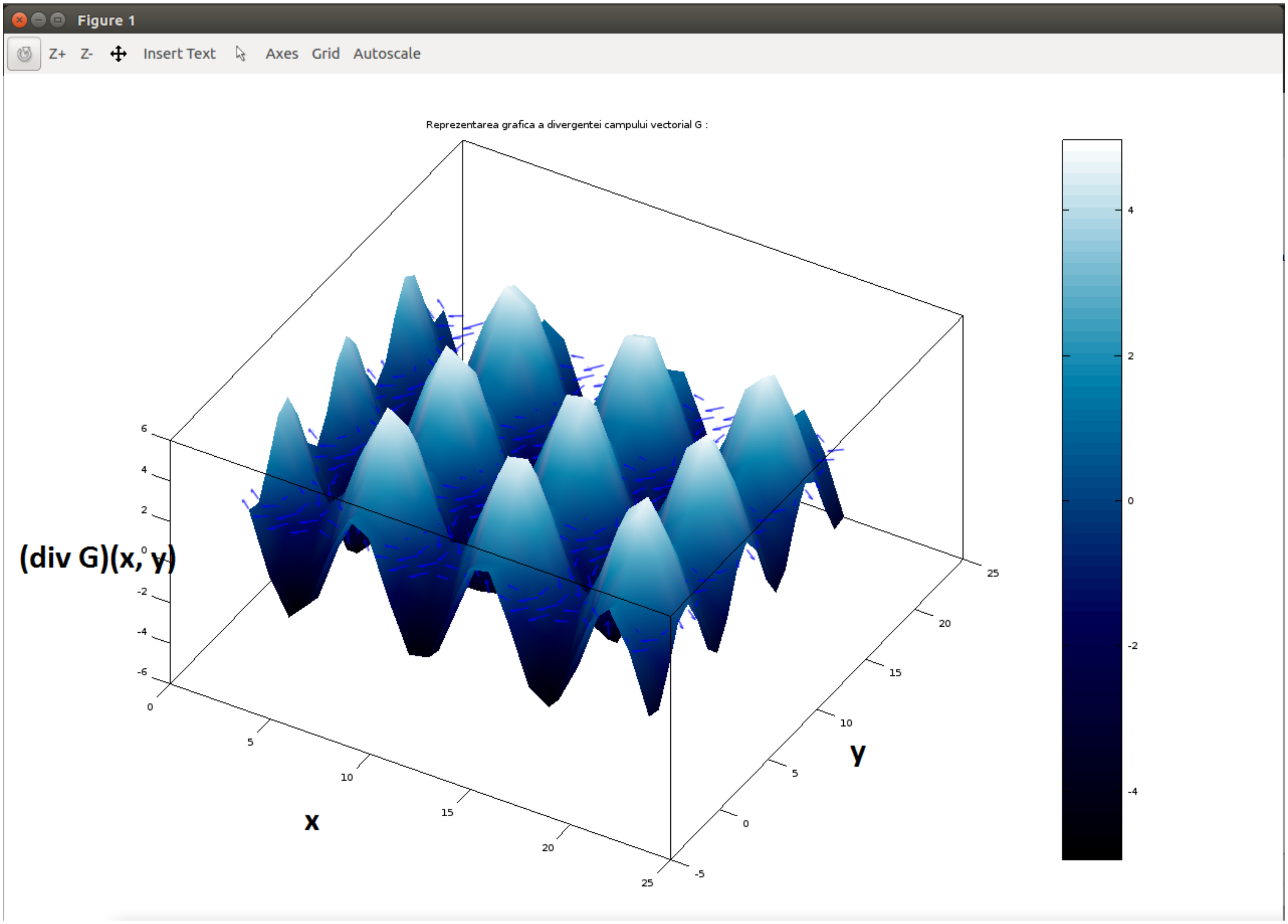

Figure 6: **Divergenta campului vectorial G (tridimensional).**

EXERCITIUL 2.c. - COD OCTAVE

```
function divergence_G ()
x = linspace(-10, 10, 21);
y = linspace(-10, 10, 21);

[X Y] = meshgrid(x, y);

figure; hold; colormap("ocean");
title("Reprezentarea grafica a divergentei campului vectorial G :");
%G = quiver(2 * X.* cos(Y), 3 * Y.* sin(X)); ————————- campul vectorial G
divergence_G = 2 * cos(Y) + 3 * sin(X);
mesh(divergence_G);
shading interp;
colorbar("EastOutSide");

endfunction
```

In continuare vom calcula rotorul campului vectorial G. In sens fizic, rotorul indica rotatia unui vector din camp, in jurul punctului din domeniu de care apartine. Vectorul rotorului este orientat dupa regula burghiului. Sensul orientarii rotorului (perpendicular pe suprafata campului) indica sensul de rotatie al vectorului, iar modulul acestuia descrie viteza de rotatie a vectorului din camp. Matematic, pentru un camp vectorial bidimensional G(x,y) = G_x(x,y) · i + G_y(x,y) · j, rotorul este egal cu: ∂G$_y$(x,y) / ∂x - ∂G$_x$(x,y) / ∂y .

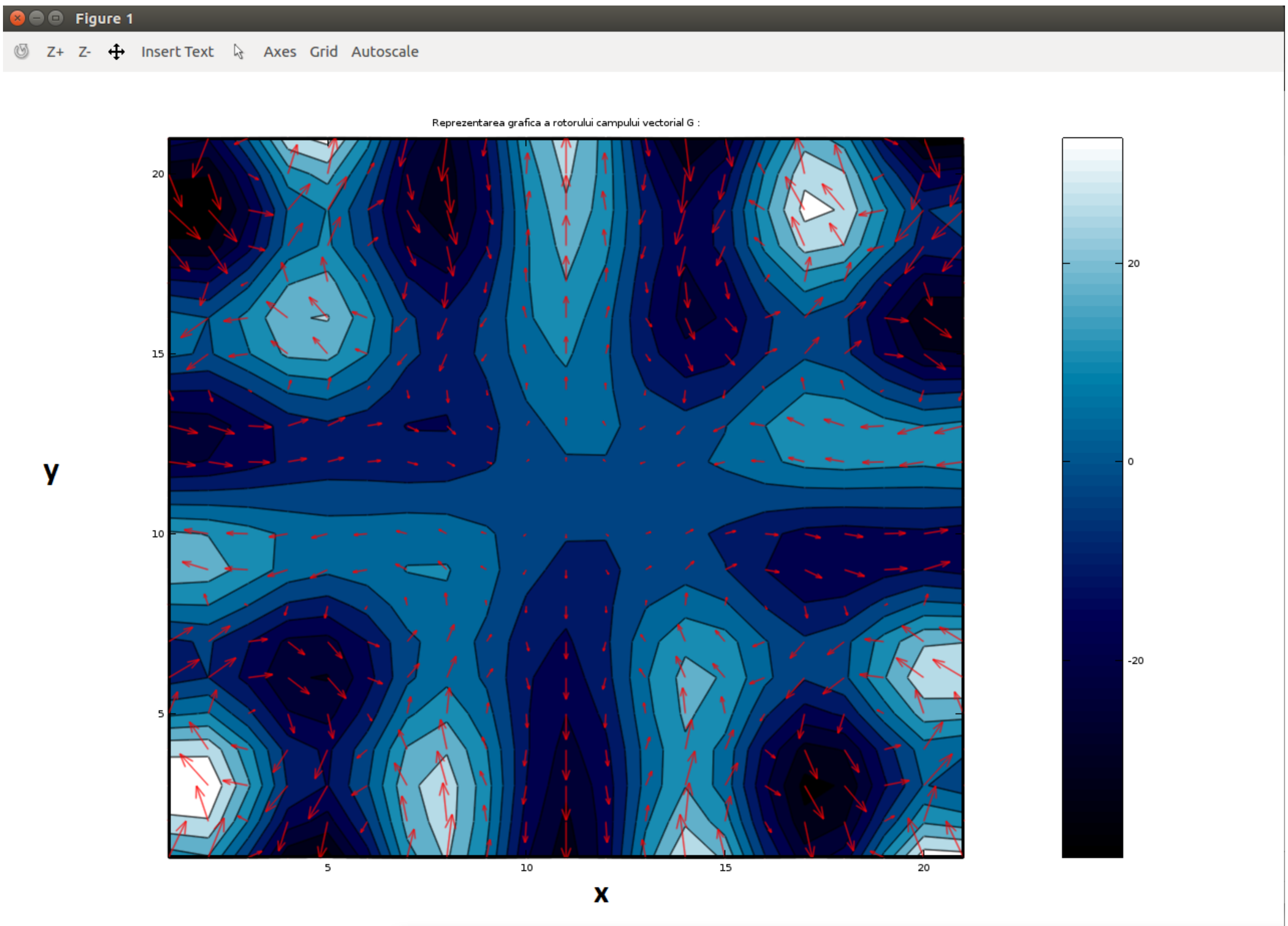

Figure 7: **Rotorul campului vectorial G (bidimensional).**

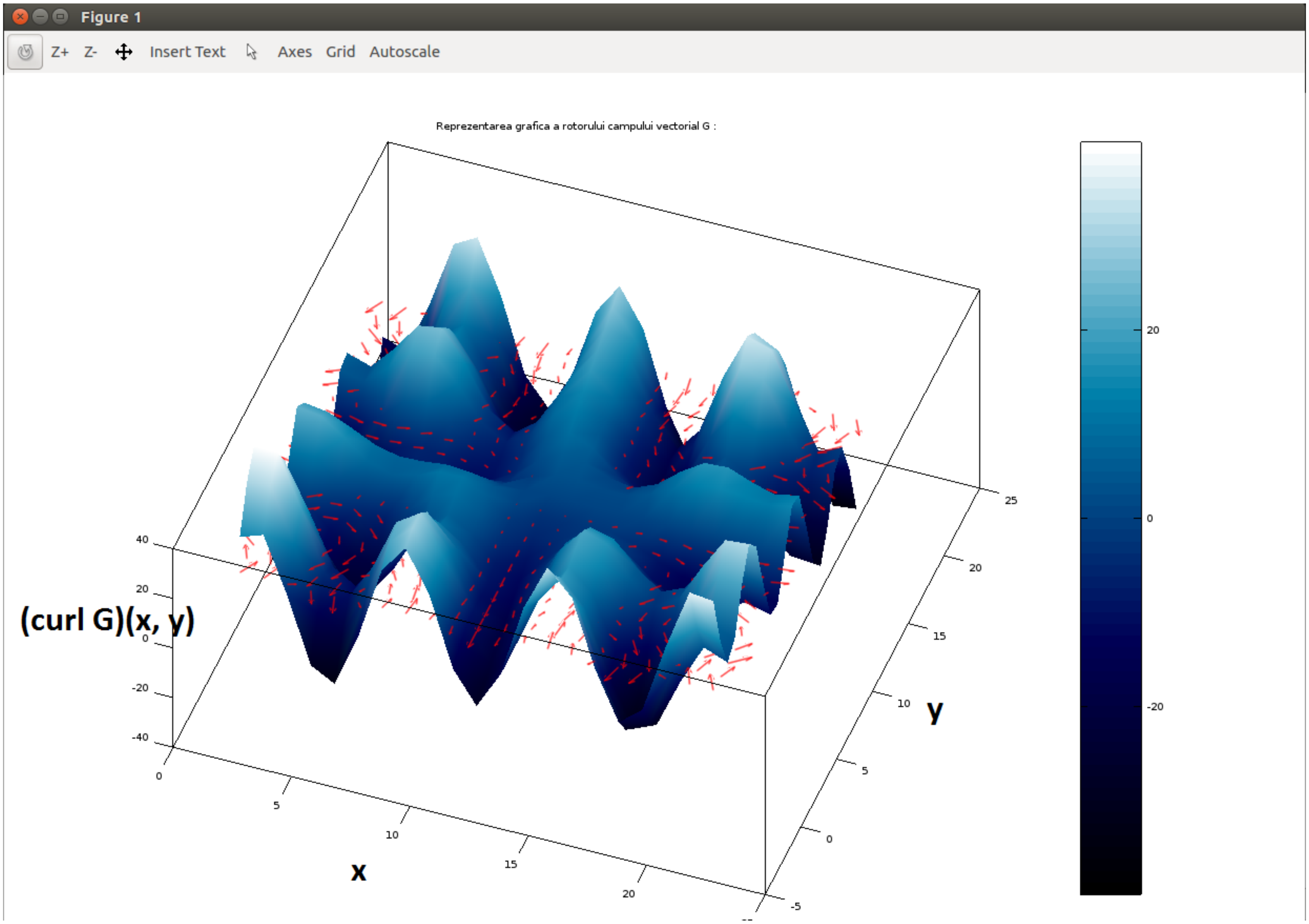

Figure 8: **Rotorul campului vectorial G (tridimensional).**

EXERCITIUL 2.d. - COD OCTAVE

```
function curl_G ()
x = linspace(-10, 10, 21);
y = linspace(-10, 10, 21);

[X Y] = meshgrid(x, y);

figure; hold; colormap("ocean");
title("Reprezentarea grafica a rotorului campului vectorial G :");
%G = quiver(2 * X.* cos(Y), 3 * Y.* sin(X),'r'); ------------------- campul vectorial G
curl_G = 3 * Y.* cos(X) + 2 * X.* sin(Y);
mesh(curl_G); --pentru a doua figura am folosit contour
quiver(-2 * X.* sin(Y), 3 * Y.* cos(X), 'r');
%contourf(curl_G); --pentru prima figura am folosit mesh
shading interp;
colorbar("EastOutSide");

endfunction
```

EXERCITII BONUS

EXERCITIUL 1
Reprezentarea campurilor scalare variabile in timp

Consideram o functie reala de 3 variabile scalare (2 variabile spatiale x si y si o variabila temporala t) f(x,y,t), unde x si y reprezinta coordonate carteziene, iar t reprezinta timpul.
Fie aceasta functie $f : [0, \pi] \times [0, 2\pi] \times \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\} \to \mathrm{R}$, $f(x,y,t) = x \cdot \sin(y - t \cdot 0.1) \cdot i + 3 \cdot y \cdot \cos(x + t \cdot 0.1) \cdot j$. Vom realiza o animatie a hartilor echivalorilor functiei, in intervalul de timp pe care este definita functia. Variabila temporala putand lua 10 valori, inseamna ca animatia va cuprinde 10 grafice pe care le vom reprezenta mai jos.

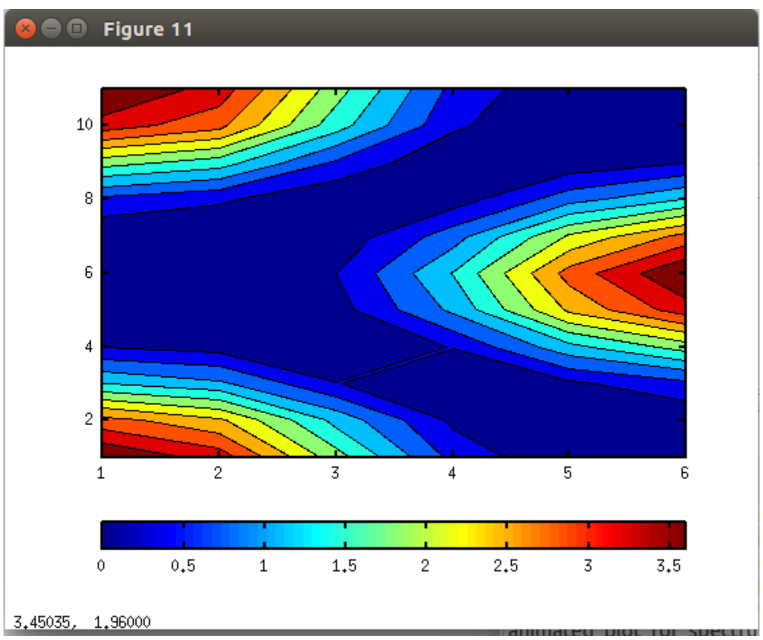

Figure 9: **Animatia 1.**

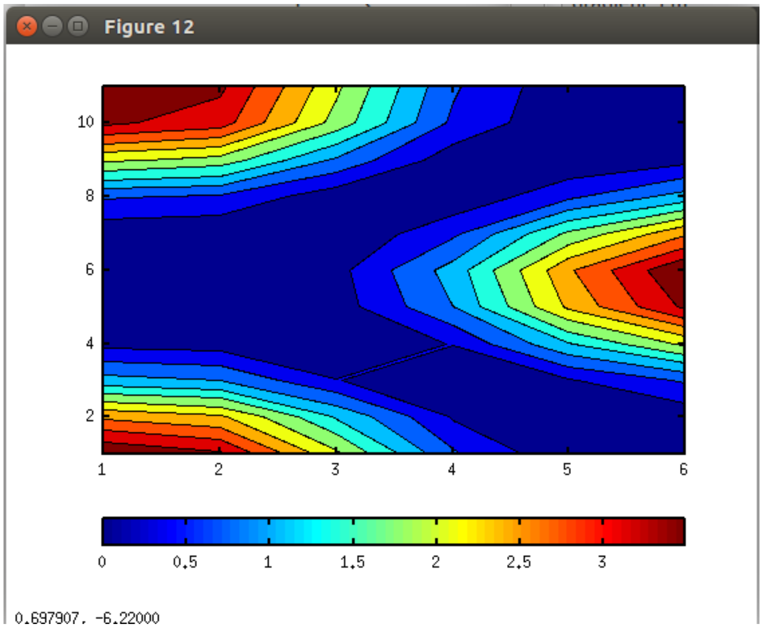

Figure 10: **Animatia 2.**

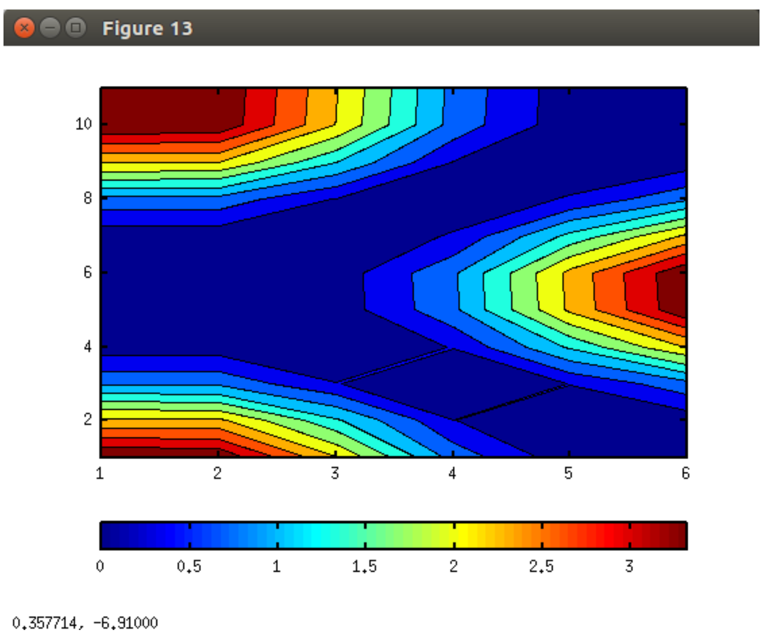

Figure 11: **Animatia 3.**

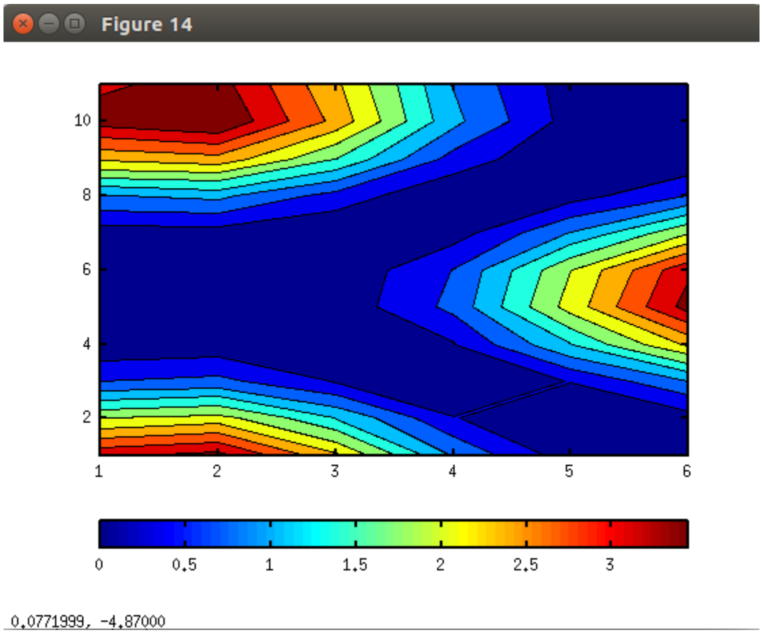

Figure 12: **Animatia 4.**

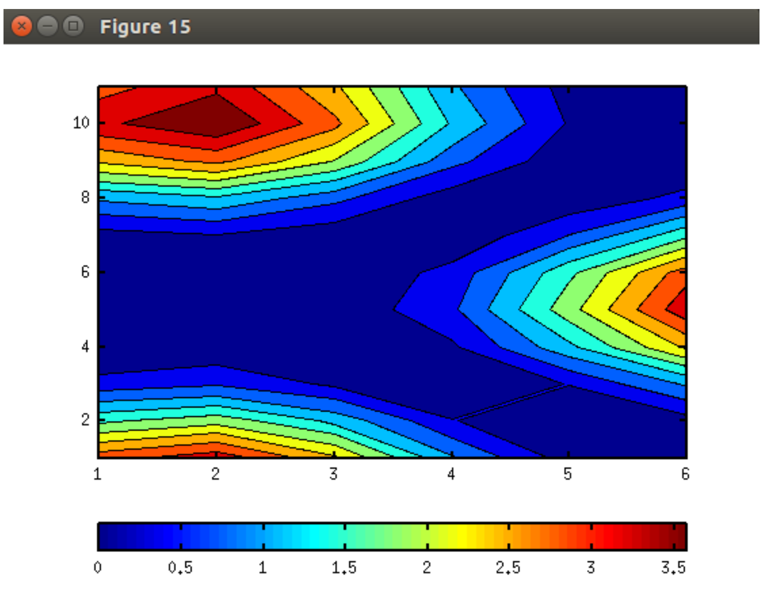

Figure 13: **Animatia 5.**

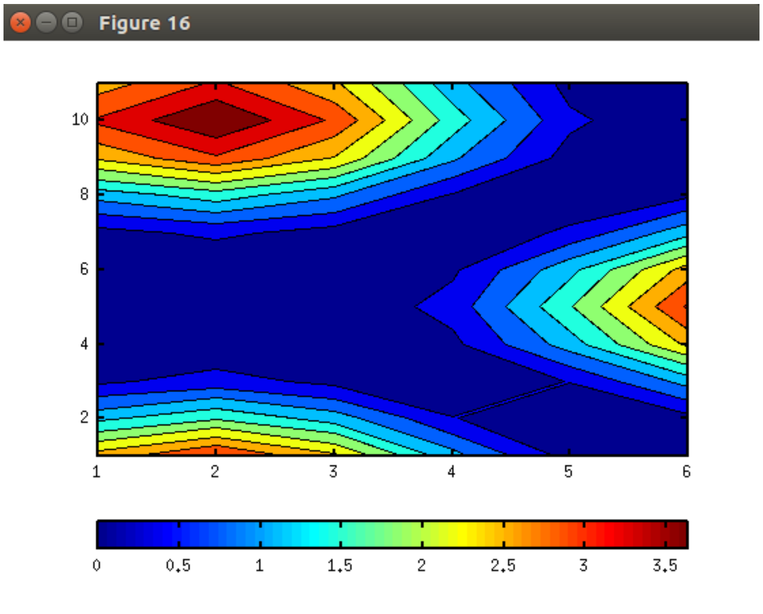

Figure 14: **Animatia 6.**

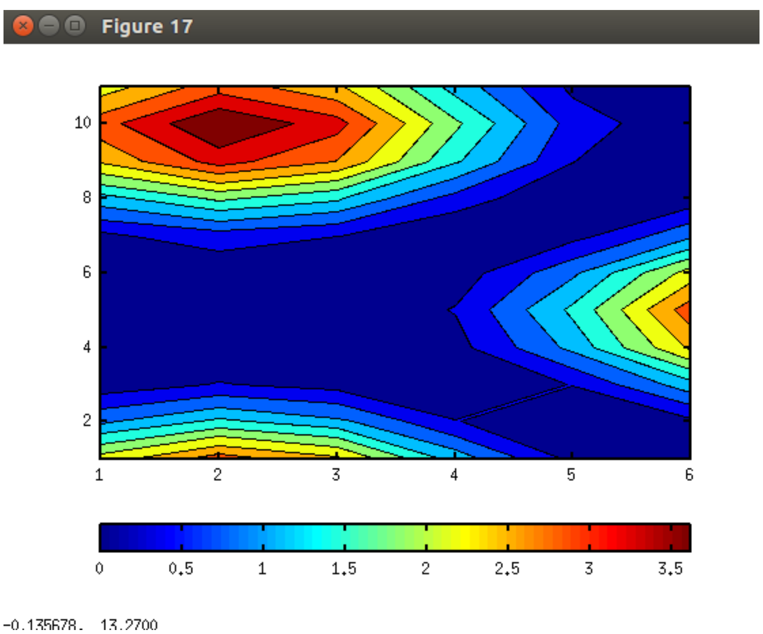

Figure 15: **Animatia 7.**

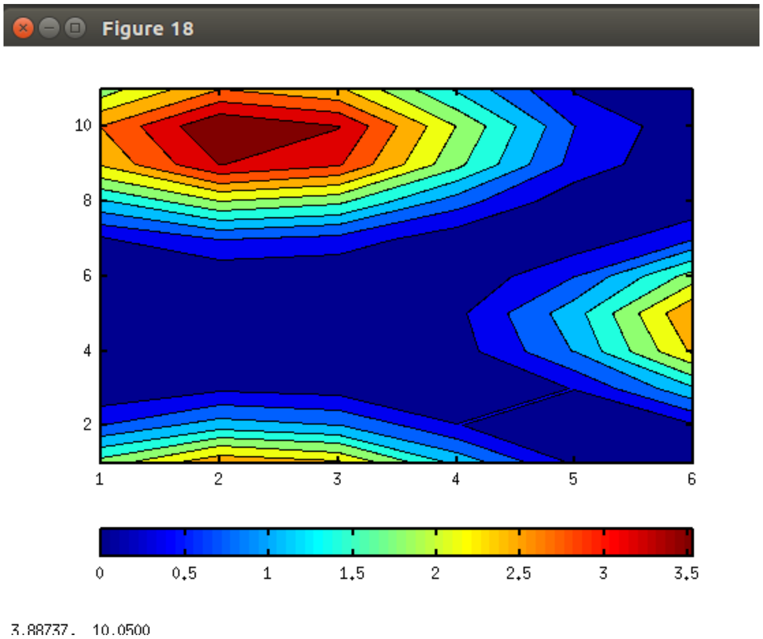

Figure 16: **Animatia 8.**

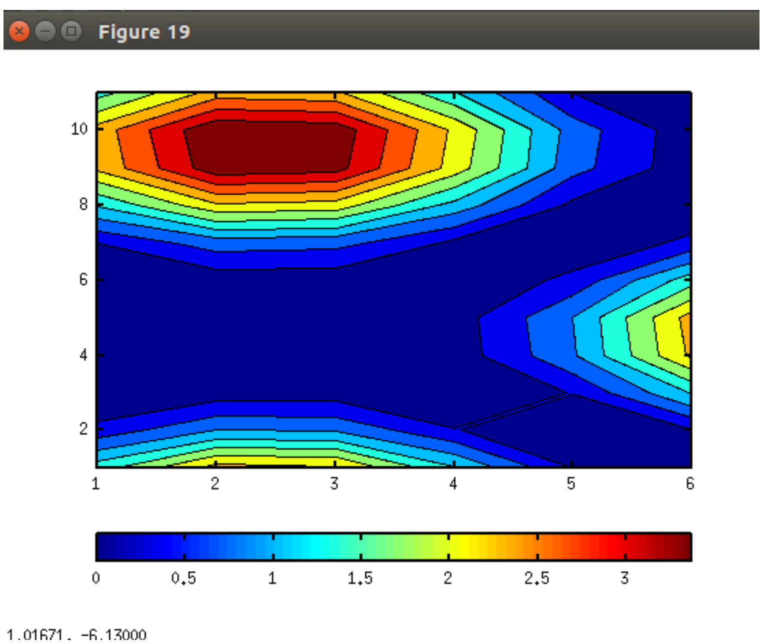

Figure 17: **Animatia 9.**

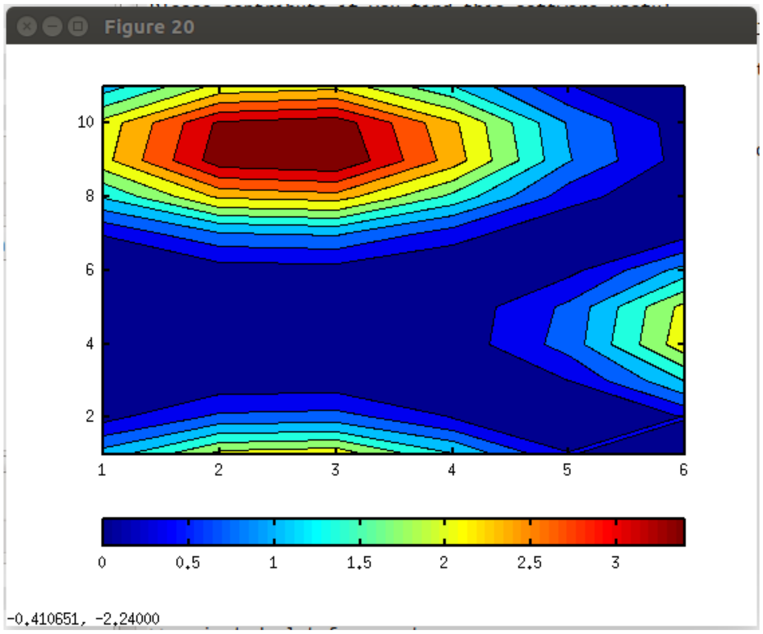

Figure 18: **Animatia 10.**

EXERCITIUL BONUS 1 - COD OCTAVE

```
function animated_plot ()
graphics_toolkit(”gnuplot”);
x = linspace(0,pi,6);
y = linspace(0,2*pi,11);
t = linspace(1,10,10);
[X Y] = meshgrid(x, y);

for i = 1 : 10
figure;
Z = (cos(X - t(i)*0.1) + cos(Y + t(i)*0.1)).2̂;

colormap(”default”);
title(sprintf(”Reprezentari, ex1”));
contourf(Z);
Fname = sprintf(”file_%i.png”, 1, ”;;”);
print(Fname);
colorbar(”SouthOutSide”);
endfor

endfunction
```

EXERCITII BONUS

EXERCITIUL 2

Reprezentarea campurilor vectoriale variabile in timp

Consideram o functie vectoriala de 3 variabile scalare (2 variabile spatiale x si y si o variabila temporala t) f(x,y,t), unde x si y reprezinta coordonate carteziene, iar t reprezinta timpul.
Fie aceasta functie $f : [0, \pi] \times [0, 2\pi] \times \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\} \rightarrow \mathbf{R}$, $f(x,y,t) = x \cdot \sin(y - t \cdot 0.3) \cdot i + 3 \cdot y \cdot \cos(x + t \cdot 0.3) \cdot j$. Vom realiza o animatie a spectrului campului functiei, in intervalul de timp pe care este definita functia. Variabila temporala putand lua 10 valori, inseamna ca animatia va cuprinde 10 grafice pe care le vom reprezenta mai jos.

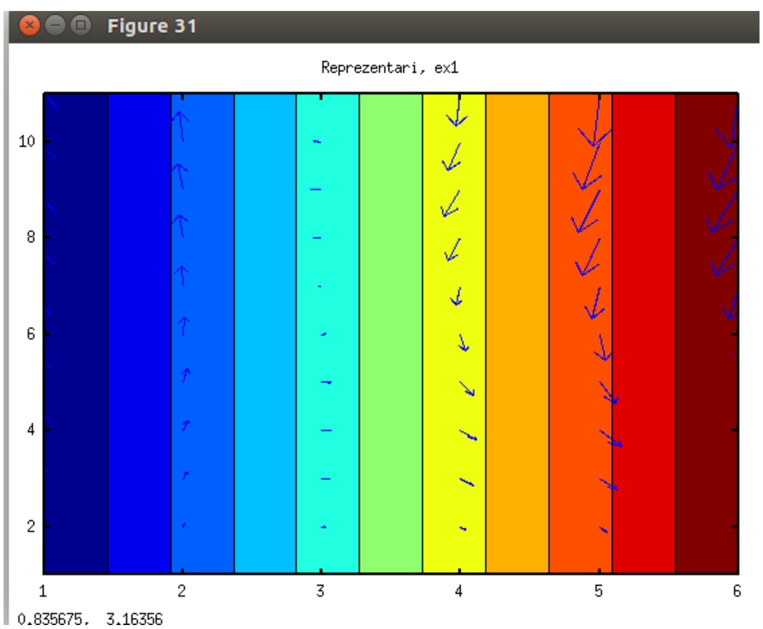

Figure 19: **Animatia 1.**

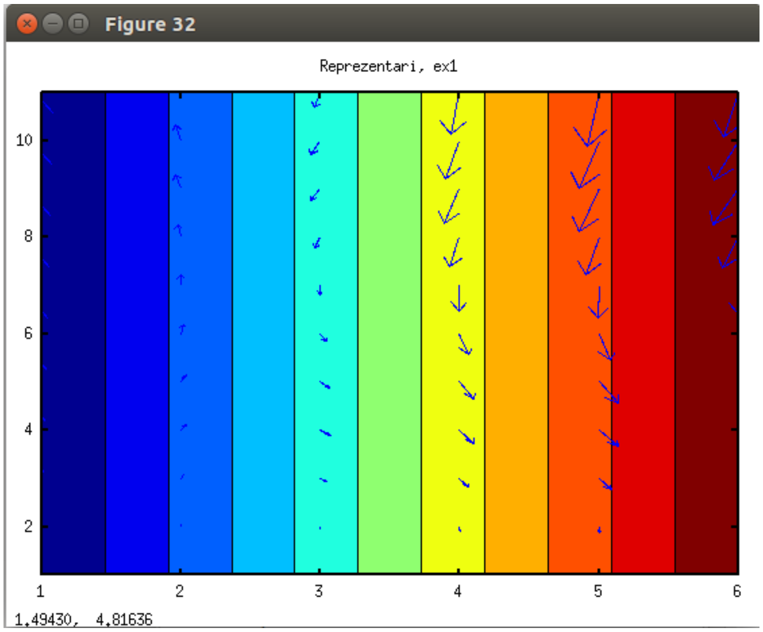

Figure 20: **Animatia 2.**

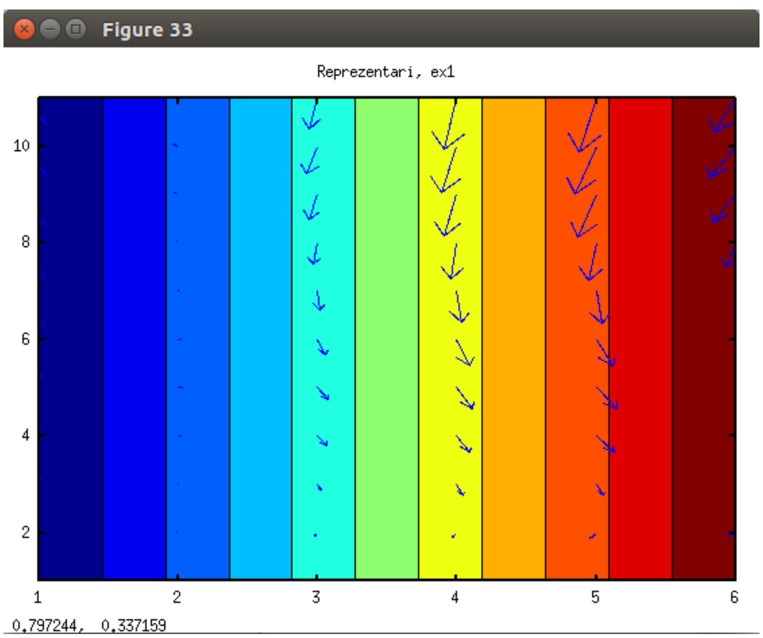

Figure 21: **Animatia 3.**

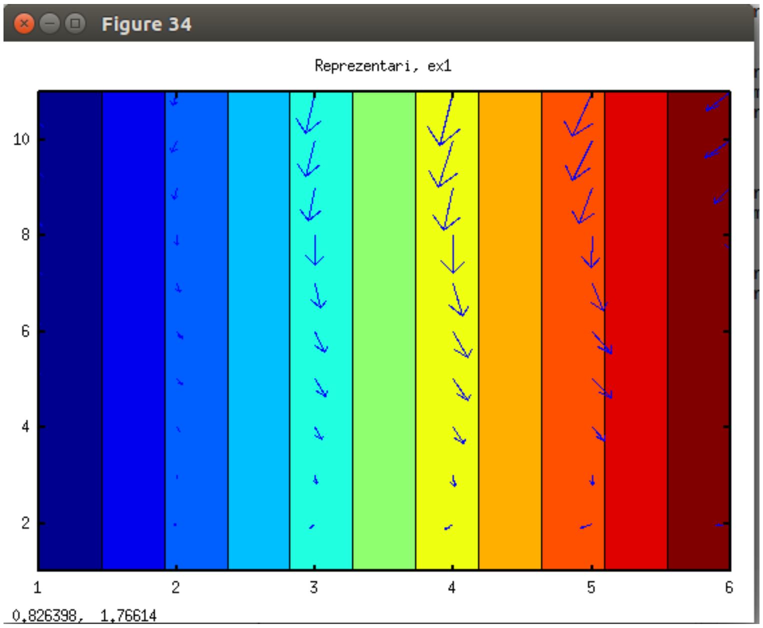

Figure 22: **Animatia 4.**

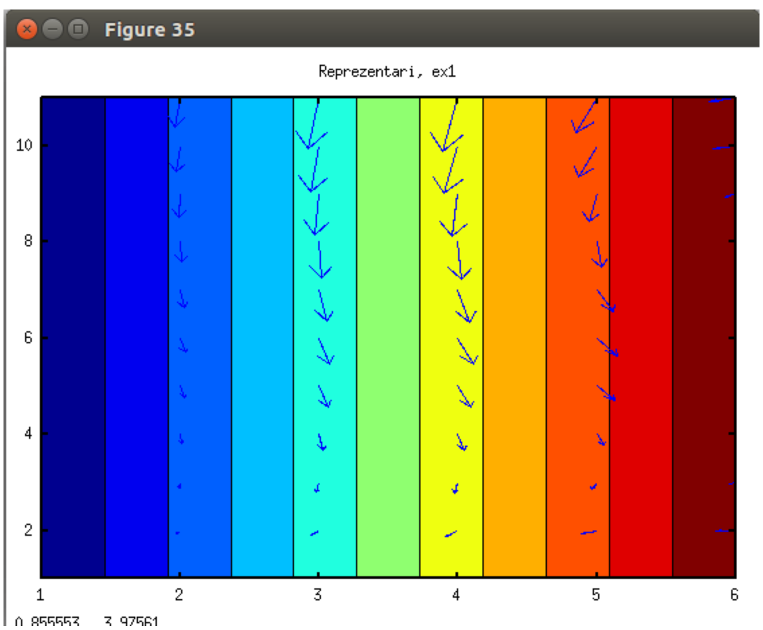

Figure 23: **Animatia 5.**

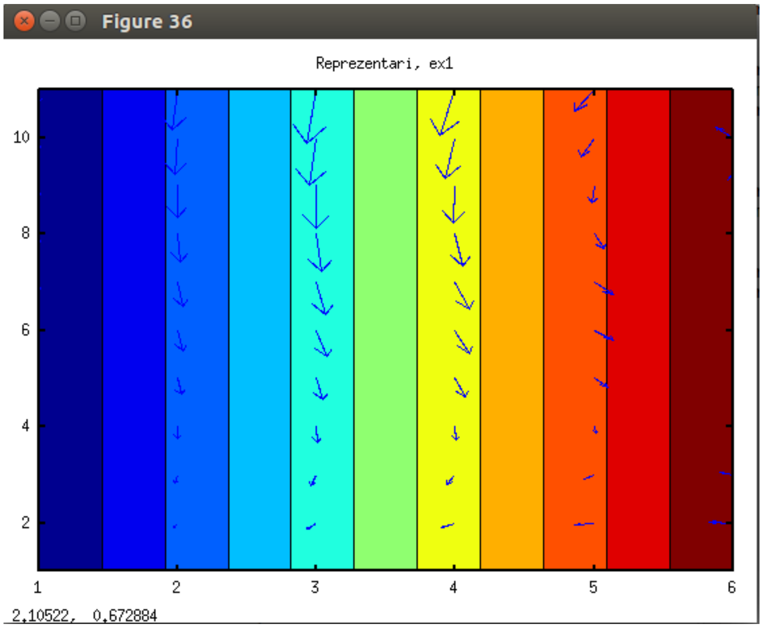

Figure 24: **Animatia 6.**

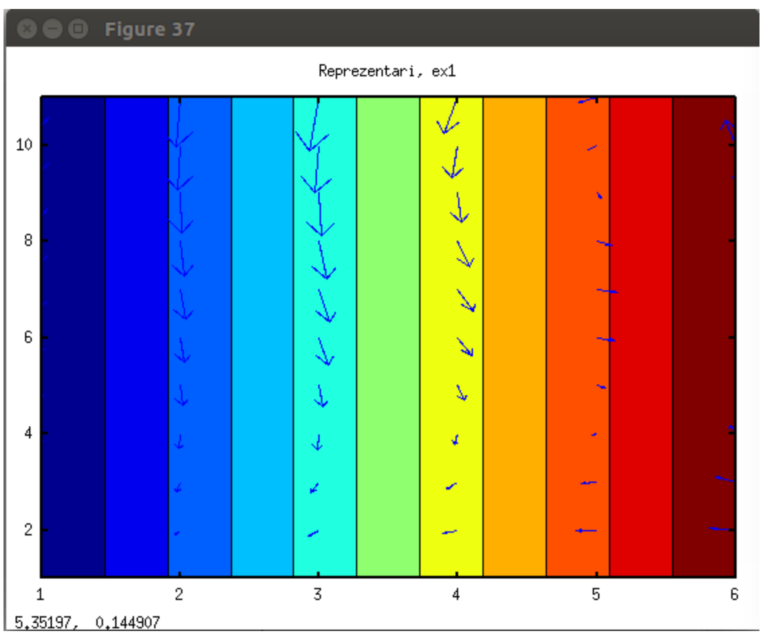

Figure 25: **Animatia 7.**

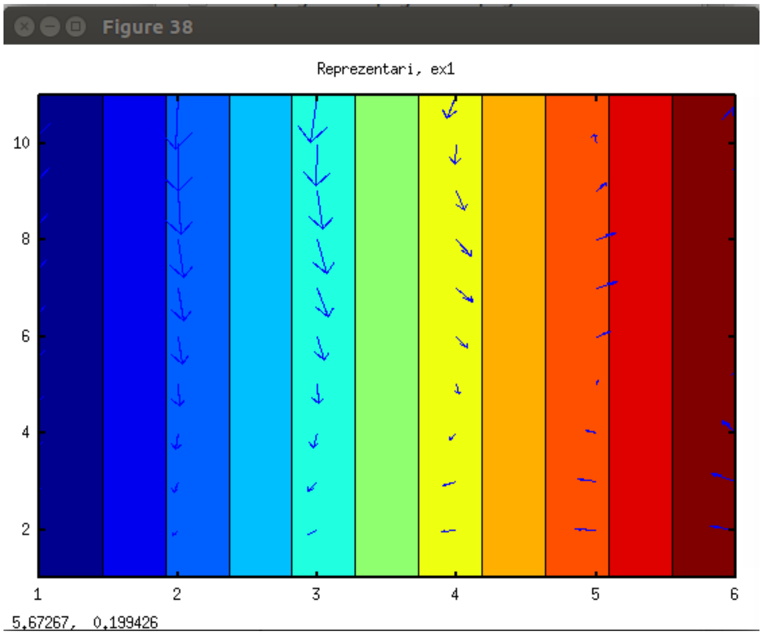

Figure 26: **Animatia 8.**

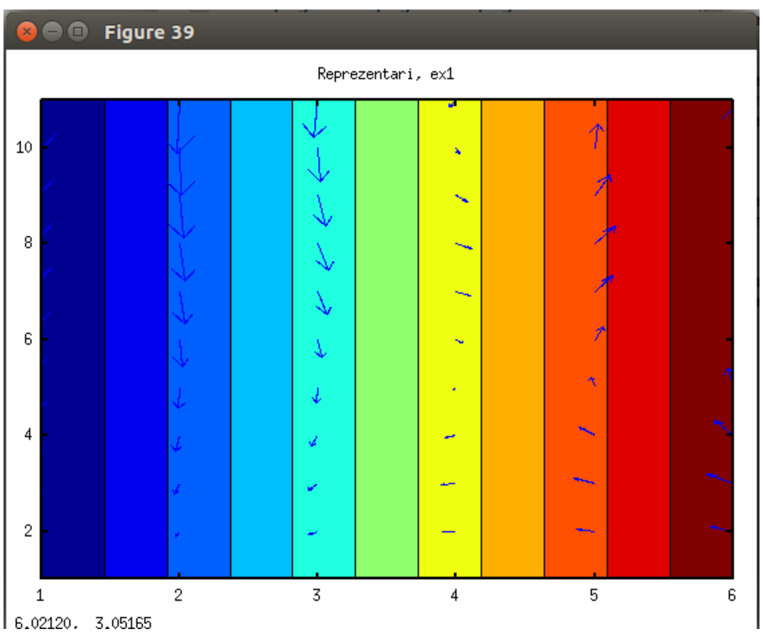

Figure 27: **Animatia 9.**

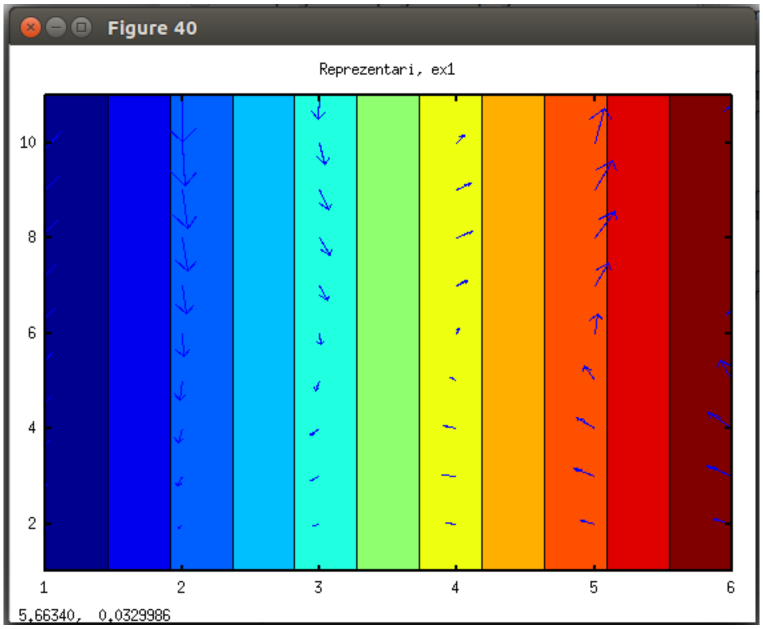

Figure 28: **Animatia 10.**

EXERCITIUL BONUS 2 - COD OCTAVE

```
    function animated_plot_for_spectrum ()
    graphics_toolkit("gnuplot");
x = linspace(0,pi,6);
y = linspace(0,2*pi,11);
t = linspace(1,10,10);
[X Y] = meshgrid(x, y);
for i = 1 : 10
figure; hold;
colormap("default");
title(sprintf("Reprezentari, ex1"));
contourf(X);
G = quiver(X.* sin(Y - t(i)*0.3), 3 * Y.* cos(X + t(i)*0.3), 'b');
Fname = sprintf("file_%i.png", i, ";;"); print(Fname);
endfor

    endfunction
```

REFERINTE