<< Algebra`InequalitySolve`

Acest pachet propune o functie pentru rezolvarea sistemelor de inegalitati. `InequalitySolve[`*expr*`,` *x*`]` gaseste conditiile pe care trebuie sa le satisfaca valorile reale x pentru ca *expr* sa fie adevarata.

```
InequalitySolve[Abs[x - 1] (x ^ 2 - 3) > 3, x]
```

$$x < -2 \,||\, x > \frac{1}{2}\left(1 + \sqrt{13}\right)$$

```
InequalitySolve[x ^ 2 - x - 1 > 0, x]
```

$$x < \frac{1}{2}\left(1 - \sqrt{5}\right) \,||\, x > \frac{1}{2}\left(1 + \sqrt{5}\right)$$

```
InequalitySolve[(x ^ 2 - 2 * x + 1) / (x ^ 2 + 3 * x - 2) > 0, x]
```

$$x < \frac{1}{2}\left(-3 - \sqrt{17}\right) \,||\, \frac{1}{2}\left(-3 + \sqrt{17}\right) < x < 1 \,||\, x > 1$$

```
InequalitySolve[2 * x / Abs[x - 1] ≥ 0 && (1 / x) < 3 * x + 1, x]
```

$$\frac{1}{6}\left(-1 + \sqrt{13}\right) < x < 1 \,||\, x > 1$$

<< DiscreteMath`RSolve`

```
RSolve[{a[n] == n * a[n - 1], a[1] == 1}, a[n], n]
```

$$\{\{a[n] \to n!\}\}$$

```
RSolve[{a[n] == a[n - 1] + 3 a[n - 2], a[0] == a[1] == 1}, a[n], n]
```

$$\left\{\left\{a[n] \to \frac{2^{-1-n}\left(-\left(1 - \sqrt{13}\right)^{1+n} + \left(1 + \sqrt{13}\right)^{1+n}\right)}{\sqrt{13}}\right\}\right\}$$

<< Algebra`AlgebraicInequalities`

Pachetul pune in evidenta existenta unei functii utile in rezolvarea sistemelor de inegalitatilor polinomiale in una sau mai multe necunoscute. `SemialgebraicComponents[`*inecuatii*`,` *var*`]` va returna un set finit de solutii ale sistemului de inegalitati.Variabila *inecuatii este o lista de inegalitati tari* , in care in ambele parti ale inegalitatiloris se gasesc polinoame in variabila *var* cu coeficienti rationali.

```
SemialgebraicComponentInstances[{(x - 1) (x^2 - 3) > 3}, x]
```

$$\left\{-\frac{1}{8}, 3\right\}$$

```
SemialgebraicComponentInstances[{x (x^2 - 1) (x^2 - 2) > 0}, x]
```

$$\left\{-\frac{9}{8}, \frac{219}{256}, 3\right\}$$

```
SemialgebraicComponentInstances[{x^2 - y^2 < 2, x y > 1}, {x, y}]
```

$$\left\{\{-2, -3\}, \left\{-\frac{3}{2}, -1\right\}, \left\{-\frac{3}{16}, -6\right\}, \{1, 2\}, \left\{\frac{3}{2}, 1\right\}, \{2, 3\}\right\}$$

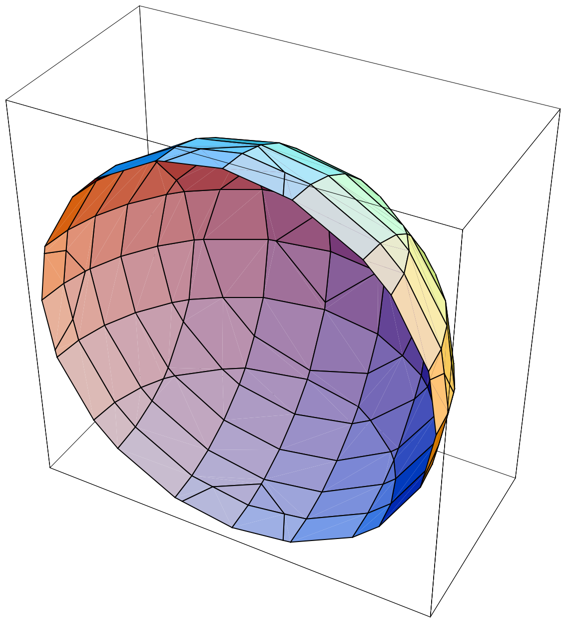
SemialgebraicComponentInstances[

$$\left\{x^2 - \frac{y^2}{4} + \frac{z^2}{9} > 1, x^2 + (y+1)^2 + (z-2)^2 < \frac{1}{9}\right\}, \{x, y, z\}]$$

{}

<< Graphics`ContourPlot3D`

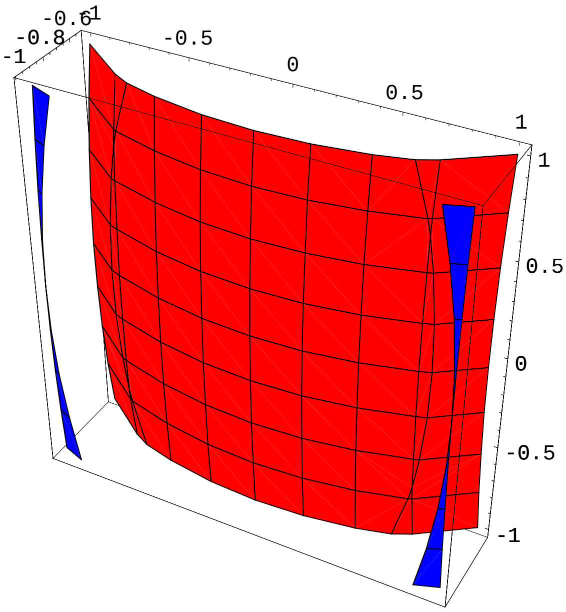
```
ContourPlot3D[Cos[Sqrt[x ^ 2 + y ^ 2 + z ^ 2]], {x, -2, 2}, {y, 0, 2}, {z, -2, 2}]
```



- Graphics3D -

```
data = Table[x ^ 2 - 2 * y ^ 3 + (1 / 5) * z ^ 2,
      {z, -1, 1, .25}, {y, -1, 1, .25}, {x, -1, 1, .25}];
```

```
ListContourPlot3D[data, DataRange → {{-1, 1}, {-1, 1}, {-1, 1}},
  Contours → {1.5`, 3.`}, Lighting → None, Axes → True,
  ContourStyle → {{RGBColor[0, 0, 1]}, {RGBColor[1, 0, 0]}}]
```



- Graphics3D -

<<Calculus`DSolveIntegrals` utilizat in obtinerea integralelor complete ale ecuatiilor cu derivate partiale
CompleteIntegral[eqn,u[x,y,..],{x,y,..}] Construieste integrala completa a ecuatiei diferentiale in raport cu u(x,y,..)

<< Calculus`DSolveIntegrals`

```
CompleteIntegral[
  Derivative[0, 1][u][x, y] ^ 2 == (u[x, y] + x ^ 3 * Derivative[1, 0][u][x, y] ^ 1/2)/y,
  u[x, y], {x, y}, IntegralConstants → C]
```

$$\left\{\left\{u[x, y] \to y + \sqrt{y}\ C[1] + \frac{1}{4} C[1]\left(4\, e^{\frac{1}{x^2}} + C[1]\right)\right\}\right\}$$

```
CompleteIntegral[
  u[x, y] – (1 + y) * Derivative[0, 1][u][x, y] + x ^ 2 * Derivative[1, 0][u][x, y] ^ 2 +
    Derivative[1, 0][u][x, y] ^ 2 == 0, u[x, y], {x, y}, IntegralConstants → F]
```

$$\left\{\left\{u[x, y] \to \frac{1}{4}\left(-\text{ArcSinh}[x]^2 - 2\,\text{ArcSinh}[x]\,F[1] - F[1]^2 + 4\,F[2] + 4\,y\,F[2]\right)\right\}\right\}$$

In mecanica analitica una din problemele importante este cea a determinarii integralelor prime sau constantelor de miscare. Pachetul permite determinarea invariantilor diferentiali pentru un sistem de ecuatii diferentiale in termenii variabilelor u[x],v[x],... si x

DifferentialInvariants[{eq1,eq2,..},{u[x],v[x],..},x]

```
DifferentialInvariants[{u'[x] == –(u[x] (u[x] + v[x])), v'[x] == v[x] (u[x] + v[x])}, {u, v}, x]
```

$$\left\{\sqrt{u\,v}\;x + \text{ArcTan}\left[\frac{u}{\sqrt{u\,v}}\right],\; u\,v\right\}$$

```
<< "VectorFieldPlots`"
```

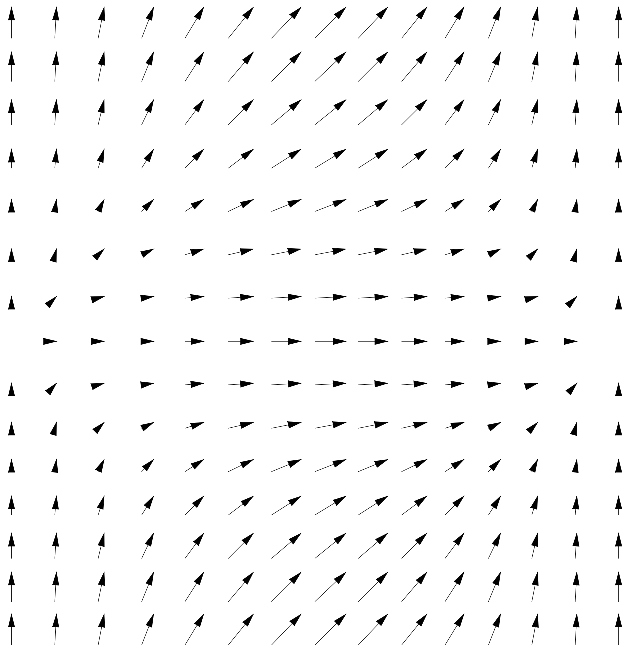
Acest pachet permite reprezentarea grafica bidimensionala a campurilor vectoriale PlotVectorField[{f_x, f_y},{x, xmin, xmax},{y, ymin, ymax}] permite reprezentarea grafica a unui camp vectorial de forma f(x,y}=$f_x \vec{i} + f_y \vec{j}$

PlotGradientField[f, {x, xmin, xmax}, {y, ymin, ymax}] pemite reprezentarea grafica a gradientului campului vectorial de forma Grad f{x,y}= $\frac{\partial f}{\partial x}\vec{i} + \frac{\partial f}{\partial y}\vec{j}$. Acest camp va da directia in care are loc cea mai rapida crestere a functiei f(x,y).

PlotHamiltonianField[f, {x, xmin, xmax}, {y, ymin, ymax}] permite reprezentarea grafica a campului vectorial Hamiltonian in care componentele sunt $\frac{\partial f}{\partial y}$ si $-\frac{\partial f}{\partial x}$. Cand functia considerata este tratata ca si Hamiltonianul unui sistem mecanic, campul vectorial Hamiltoniana va da ecuatiile de miscare in spatiul fazelor

Spre exemplu, componentele campului vectorial sunt date de Sin[x]^2 si Cos[y]^2

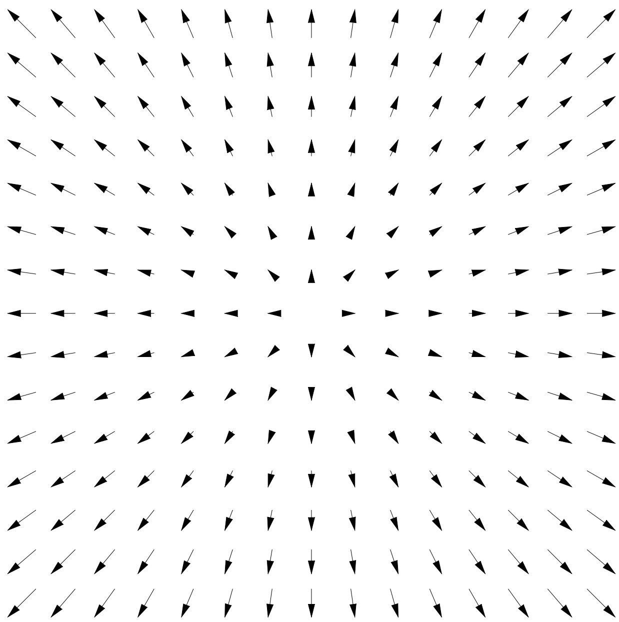
```
Needs["VectorFieldPlots`"];
VectorFieldPlots`VectorFieldPlot[{Sin[x]^2, Cos[y]^2}, {x, 0, π}, {y, 0, π}]
```



- Graphics -

Sau gradientul campului unui potential de forma x^2 + y^2:

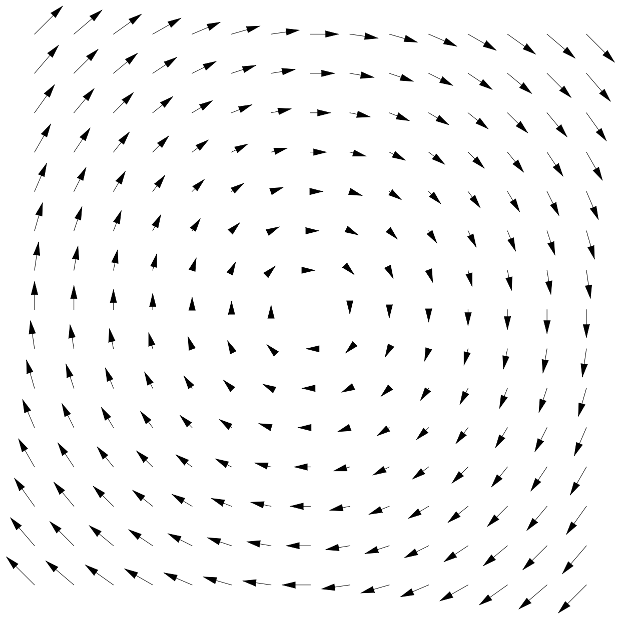
```
Needs["VectorFieldPlots`"];
VectorFieldPlots`GradientFieldPlot[x^2 + y^2, {x, -3, 3}, {y, -3, 3}]
```



- Graphics -

In plan, cele doua tipuri de campuri vectoriale sunt intotdeauna perpendiculare

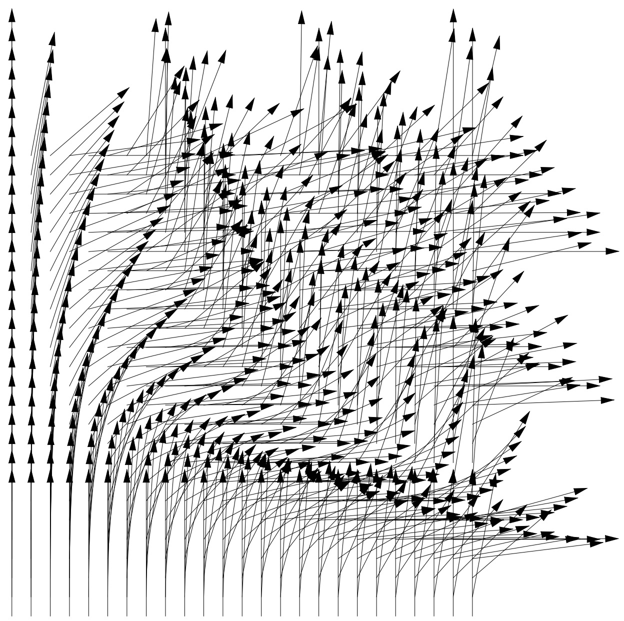
```
Needs["VectorFieldPlots`"];
VectorFieldPlots`HamiltonianFieldPlot[x^2 + y^2, {x, -3, 3}, {y, -3, 3}]
```



- Graphics -

Daca se doreste cresterea numarului de puncte reprezentative (spre exemplu 25 in fiecare directie) si reprezentarea fiecarui vector la scala de 1/2, utilizam:

```
Needs["VectorFieldPlots`"];
VectorFieldPlots`VectorFieldPlot[{Sin[x y]^2, Cos[x y]^2}, {x, 0, π}, {y, 0, π},
  PlotPoints → 25, ScaleFunction → (0.5` #1 &), ScaleFactor → None]
```



-Graphics-

Cum am controlat afisarea vectorilor?

Exista trei elemente de control al marimii vectorilor:

`ScaleFunction` care poate fi o functie pura ce ia valoare vectorului si returneaza noua marime a acestuia

`MaxArrowLength` esimina orice vector a carui lungime este mai mare decat valoarea returnata de functia anterioara

`S c a l e F a c t o r`

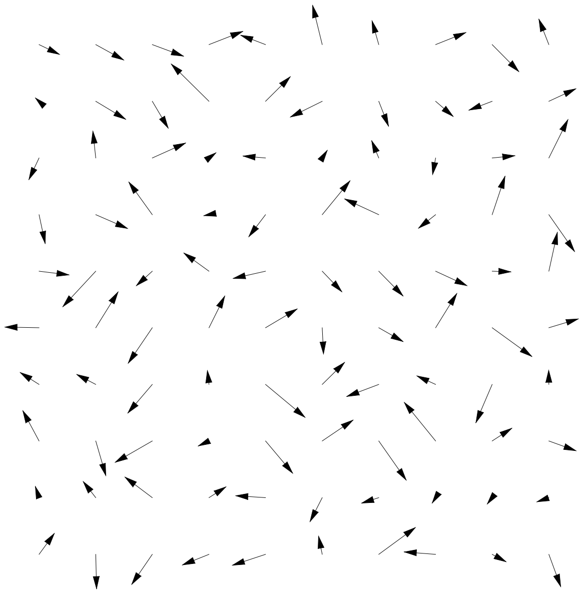
$$\begin{pmatrix} \texttt{Automatic}, & \text{sacalare fara suprapunerrea varfurilor vect. pe origea vect. adiacenti} \\ \text{None} & \text{fara scalare} \end{pmatrix}$$

Fie o retea aleatoare de vectori

```
vectaleat = Table[RandomReal[{-0.7`, 0.7`}, 2], {i, 10}, {j, 10}];
```

Sa afisam acesti vectori

```
Needs["VectorFieldPlots`"]; VectorFieldPlots`ListVectorFieldPlot[vectaleat]
```



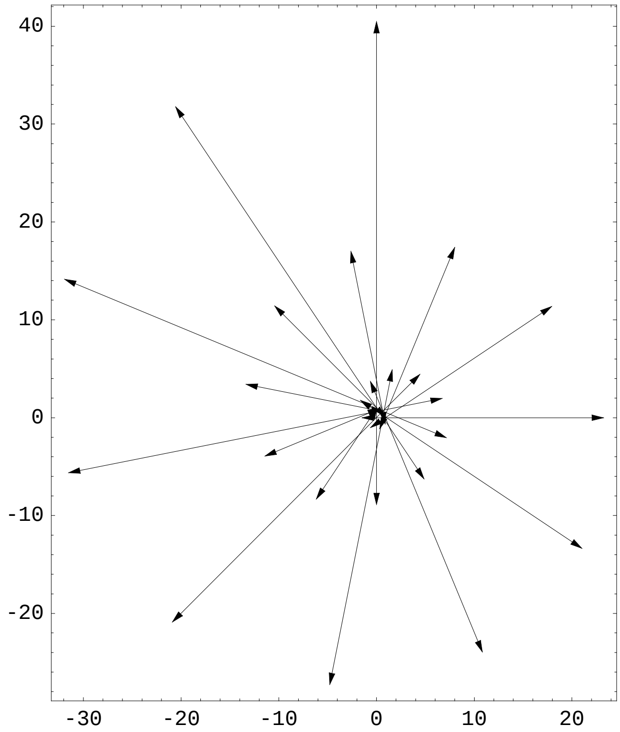
- Graphics -

Generam o lista de vectori si reprezentam vectorii fara scalare

```
vectori =
    Table[{{Sin[x] ^ 2, Cos[x] ^ 2}, {x ^ 2 Sin[3 x], x ^ 2 Cos[3 x]}}, {x, 0, 2 Pi, Pi / 16}];
```

Needs["VectorFieldPlots`"];

VectorFieldPlots`ListVectorFieldPlot[vectori, ScaleFactor → None, Frame → True]



- Graphics -