

# Lösung Beispiel 6, Helmut Hörner MatNr. 08850092

## Festlegen der Potentiale der sechs Leiter

```
In[1]:= randpot = {0, 100, 0, 0, 100, 0};
```

## Festlegen von Geometrie und Gitterweite in Metern

```
In[2]:= d1 = 0.0002  
yWidth = 25 d1  
xWidth = 4 yWidth
```

```
Out[2]= 0.0002
```

```
Out[3]= 0.005
```

```
Out[4]= 0.02
```

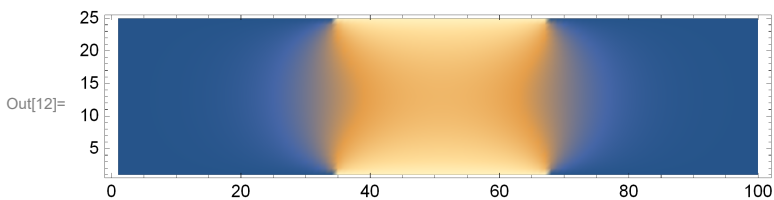
## Erzeuge Gitter (x=1..xMax, y=1...yMax)

```
In[5]:= xMax = IntegerPart[Round[xWidth/d1, 1]];  
           |ganzzahliger T...|runde  
yMax = IntegerPart[Round[yWidth/d1, 1]];  
           |ganzzahliger T...|runde  
grid = Table[  
           |Tabelle  
           If[y == 1, randpot[[IntegerPart[Round[1 + Floor[(x - 1)/(xMax/3), 1]]]],  
           |wenn |ganzzahliger T...|runde |runde ab  
           If[y == yWidth/d1,  
           |wenn  
           randpot[[IntegerPart[Round[4 + Floor[(x - 1)/(xMax/3), 1]]]], 0]],  
           |ganzzahliger T...|runde |runde ab  
           {y, 1, yMax}, {x, 1, xMax}];
```

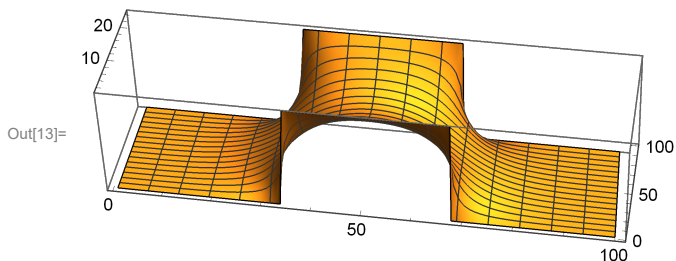
## Potentialberechnung mit Relaxationsmethode

```
In[8]:= MaxError = 0.1;
newGrid = grid;
mDiff = 1;
For[zz = 1, mDiff > MaxError, zz++, grid = newGrid; mDiff = 0;
  For-Schleife
  For[yy = 2, yy < yMax, yy++,
    For-Schleife
    For[xx = 1, xx <= xMax, xx++,
      For-Schleife
      newGrid[[yy, xx]] = N[1/4 (grid[[yy - 1, xx]] + grid[[yy + 1, xx]] +
        numerischer Wert
        grid[[yy, Max[1, xx - 1]]] + grid[[yy, Min[xMax, xx + 1]]])];
      größtes Element
      kleinstes Element
      mDiff = Max[mDiff, Abs[newGrid[[yy, xx]] - grid[[yy, xx]]];
      größtes El...
      Absolutwert
    ]
  ]
];
```

```
In[12]:= ListDensityPlot[grid, AspectRatio -> {xWidth / yWidth}]
  listenbezogene graphische ...
  Seitenverhältnis
```

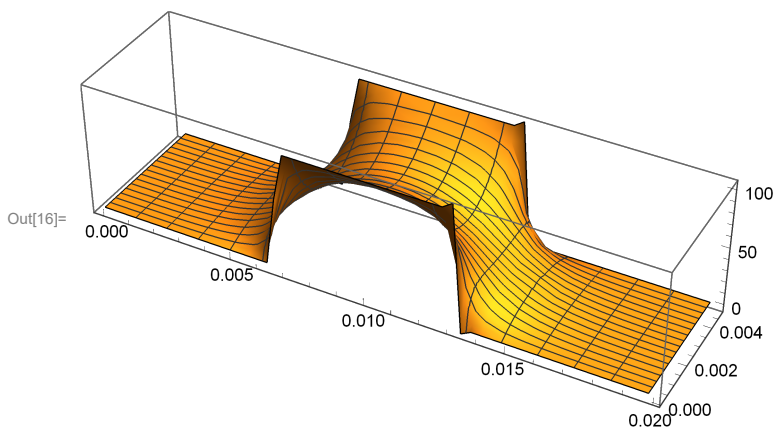


```
In[13]:= ListPlot3D[grid, BoxRatios -> {xWidth, yWidth, yWidth}]
  listenbezogene 3D-G...
  Seitenverhältnis der Box
```



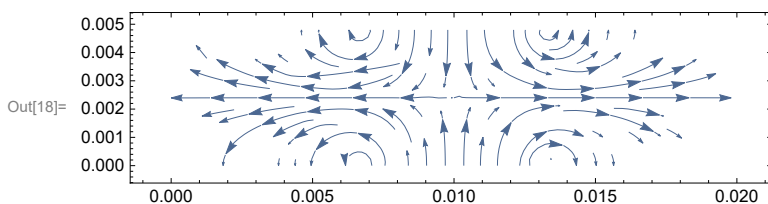
## Ermittlung einer Potentialfunktion $\phi[x, y]$ durch Interpolation

```
In[14]:=  $\phi1 = \text{ListInterpolation}[\text{grid}];$ 
          |listenbezogene Interpolation
          (* $\phi1[x, y]$  in Grid-Einheiten mit  $x=1..x\text{Max}$  und  $y=1..y\text{Max}$ *)
          |Gitter
 $\phi[x_, y_] = \phi1[(y + d1) / d1, (x + d1) / d1];$ 
          (*  $\phi[x, y]$  in Längeneinheiten (Meter) mit  $x=0..x\text{Width}-d1$  und  $y=0..y\text{Width}-d1$  *)
 $\text{Plot3D}[\phi[x, y], \{x, 0, x\text{Width} - d1\},$ 
          |stelle Funktion graphisch in 3D dar
           $\{y, 0, y\text{Width} - d1\}, \text{BoxRatios} \rightarrow \{x\text{Width}, y\text{Width}, y\text{Width}\}]$ 
          |Seitenverhältnis der Box
```



## Feldstärke $e[x, y]$ als $-\text{grad}[\phi[x, y]]$

```
In[17]:=  $e[x_, y_] = -\text{Grad}[\phi[x, y], \{x, y\}];$ 
          |Gradient
 $\text{StreamPlot}[e[x, y], \{x, 0, x\text{Width} - d1\},$ 
          |Strömungsdiagramm
           $\{y, 0, y\text{Width} - d1\}, \text{AspectRatio} \rightarrow \{x\text{Width} / y\text{Width}\}]$ 
          |Seitenverhältnis
```

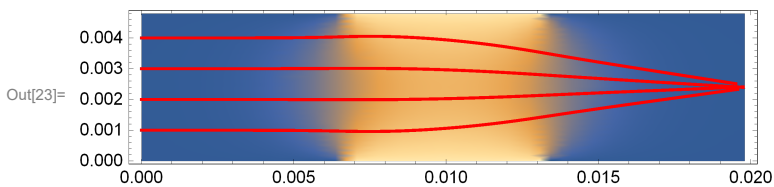


## Waagrecht einfallender Elektronenstrahl (Elektrostatische Linse)

```

In[19]= physconst = {q → 1.60217649 × 10-19, m → 9.10938215 × 10-31};
anfangsbed1 = {vx0 → 12000000, vy0 → 0, x0 → 0, y0 → (ywidth - d1) / 2};
bahnen1 =
  Table[
    NDSolve[{m x''[t] == e[x[t], y[t]][[1]] q,
      x'[0] == vx0, x[0] == x0, my''[t] == e[x[t], y[t]][[2]] q,
      y'[0] == vy0, y[0] == ywidth / 5 * i} /. physconst /. anfangsbed1,
      {x, y}, {t, 0, 1.7 × 10-9},
      {i, 1, 4}];
bahnplots1 =
  Table[
    ParametricPlot[{bahnen1[[i, 1, 1, 2]][t], bahnen1[[i, 1, 2, 2]][t]}, {t, 0,
      1.7 × 10-9}, PlotStyle → Red, PlotRange → {{0, xwidth - d1}, {0, ywidth - d1}},
      {i, 1, 4}];
Show[DensityPlot[φ[x, y], {x, 0, xwidth - d1},
  {y, 0, ywidth - d1}, AspectRatio → {xwidth / ywidth}], bahnen1]

```

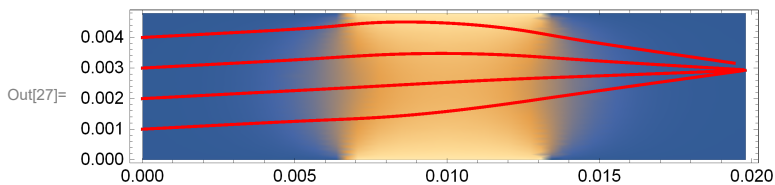


## Einfallender Elektronenstrahl mit Ablenkung nach oben

```

In[24]= anfangsbed2 = {vx0 -> 12000000, vy0 -> 630000, x0 -> 0, y0 -> (ywidth - d1) / 2};
bahnen2 =
  Table[
    Tabelle
    NDSolve[{m x''[t] == e[x[t], y[t]][[1]] q,
      löse Differentialgleichung numerisch
      x'[0] == vx0, x[0] == x0, m y''[t] == e[x[t], y[t]][[2]] q,
      y'[0] == vy0, y[0] == ywidth / 5 * i} /. physconst /. anfangsbed2,
      {x, y}, {t, 0, 1.7 * 10^-9}],
    {i, 1, 4}];
bahnplots2 =
  Table[
    Tabelle
    ParametricPlot[{bahnen2[[i, 1, 1, 2]][t], bahnen2[[i, 1, 2, 2]][t]}, {t, 0,
      parametrische Darstellung
      1.7 * 10^-9}, PlotStyle -> Red, PlotRange -> {{0, xwidth - d1}, {0, ywidth - d1}},
      Darstellungsstil rot Koordinatenbereich der Graphik
      {i, 1, 4}];
Show[DensityPlot[phi[x, y], {x, 0, xwidth - d1},
  zeige graphische Darstellung der Dichte
  {y, 0, ywidth - d1}, AspectRatio -> {xwidth / ywidth}], bahnenplots2]
Seitenverhältnis

```



## Elektronenstrahl Interaktiv

```

In[28]= Manipulate[bahnen3 =
  [manipuliere
    Table[
      [Tabelle
        Quiet[NDSolve[{m x''[t] == e[x[t], y[t]] [[1]] q, x'[0] ==  $\sqrt{\frac{\text{Abs}[VX] \text{Abs}[q]}{m}}$ ,
          [unter... [löse Differentialgleichung numerisch
            x[0] == x0, m y''[t] == e[x[t], y[t]] [[2]] q, y'[0] ==  $\text{Sign}[VY] \sqrt{\frac{\text{Abs}[VY] \text{Abs}[q]}{m}}$ ,
          [Vorzeichen
            y[0] == ywidth/5 * i} /. physconst /. anfangsbed2,
            {x, y}, {t, 0, 1.7 * 10-9}}],
            {i, 1, 4}];
        bahnplots3 =
          Table[
            [Tabelle
              ParametricPlot[{bahnen2[[i, 1, 1, 2]][t], bahnen3[[i, 1, 2, 2]][t]}, {t, 0,
                [parametrische Darstellung
                  1.7 * 10-9}, PlotStyle -> Red, PlotRange -> {{0, xwidth - d1}, {0, ywidth - d1}},
                [Darstellungsstil [rot [Koordinatenbereich der Graphik
                  {i, 2, 3}];
              Show[DensityPlot[phi[x, y], {x, 0, xwidth - d1},
                [graphische Darstellung der Dichte
                  {y, 0, ywidth - d1}, AspectRatio -> {xwidth/ywidth}], bahnplots3],
                [Seitenverhältnis
                  {{VX, 409, "X0 Energy [eV]"}, 100, 750},
                  {{VY, 0, "Y0 Energy [eV]"}, -6, 6}, ContinuousAction -> False]
                [fortlaufende Aktion [falsch

```

Out[28]=

