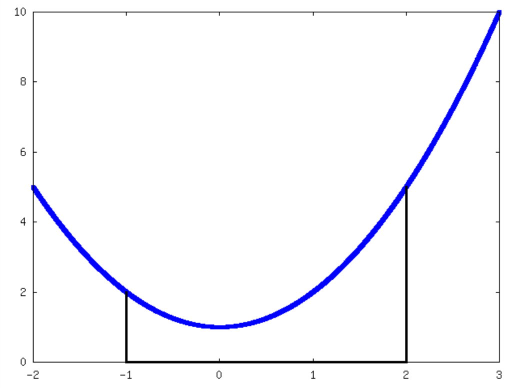
Přednáška č. 4

Téma:

1. Numerická integrace (obsah plochy)
2. Řešení ODR pomocí Eulerovy metody.
3. Aproximace okrajové úlohy, sestavení soustavy rovnic.

Numerická integrace



Vstupní data:

- Funkce $f(x)$
- Interval (a, b)

Výstupní data: Aproximace

$$\int_a^b f(x)dx \approx S$$

Nahradíme integraci na intervalu (a,b) součtem ...

- Vycházíme z definice Riemannova integrálu.
- Pro speciální funkce lze integrál tímto způsobem vyhodnotit přesně.
- Nahrazujeme přesnou hodnotu nepřesnou ... jakou děláme chybu a co o ní lze říci?

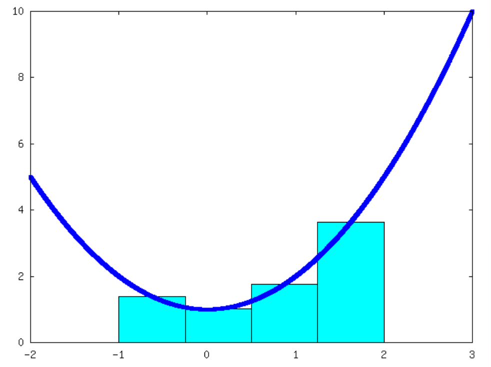
$$\int_a^b f(x)dx \approx \Delta x \frac{f(a)+f(b)}{2}$$

$$\int_a^b f(x)dx \approx \Delta x f(\frac{a+b}{2})$$

$$\int_a^b f(x)dx \approx \frac{\Delta x}{6}\left(f(a)+4f(\frac{a+b}{2})+f(b)\right)$$

- Složené pravidlo ...

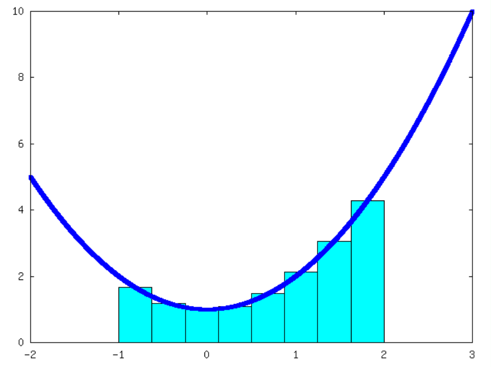
Obdelnikove pravidlo (n=4)



Přibližná hodnota S (obsah obdélníků)

$$S = \int_a^b f(x)dx \approx \sum_{\text{vsechny obdelniky}} S_k$$

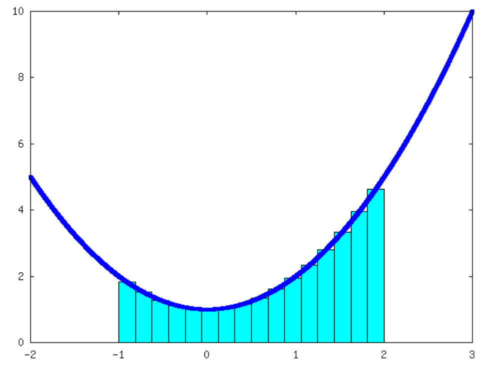
Obdelnikove pravidlo (n=8)



Přibližná hodnota S (obsah obdélníků)

$$S = \int_a^b f(x)dx \approx \sum_{\text{vsechny obdelniky}} h_k f(x_{stred,k})$$

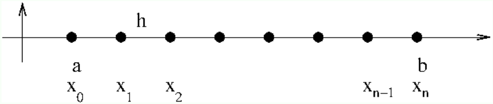
Obdelnikove pravidlo (n=16)



Přibližná hodnota S (obsah obdélníků)

$$S = \int_a^b f(x)dx \approx \sum_{\text{vsechny obdelniky}} h_k f(x_{stred,k})$$

Obdelnikove pravidlo - deleni intervalu



Vstup: n - pocet dilku

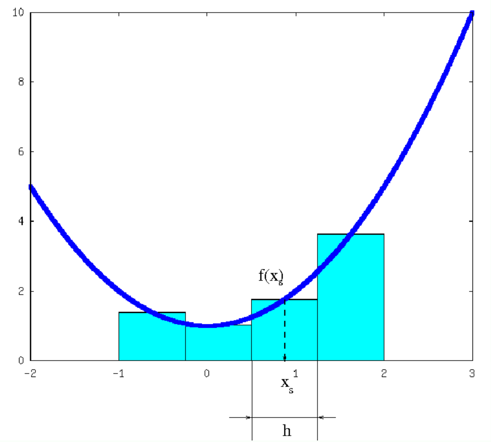
$$x_0 = a, \qquad x_n = b$$

Ekvidistantni deleni: h - krok deleni

$$x_k = a + k\,h, \qquad b = a + n\,h$$

$$h = (b - a)/n$$

Obdelnikove pravidlo (n=4)



Přibližná hodnota S (obsah obdélníků)

$$S = \int_a^b f(x)dx \approx \sum_{\text{vsechny obdelniky}} hf(x_s)$$

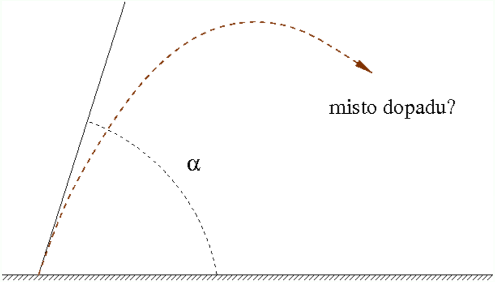
Program c. 1: Numericka integrace

```
float fce(float x)
{
  return (x*x+1.);
}
float integral(float a,float b, int n)
{
  float h,S,x;
  int i;
  S=0;  h=(b-a)/n;
  for(i=0;i<n;i++)
  {
    x=a+(i+0.5)*h;
    S=S+h*fce(x);
  }
  return S;
}
```

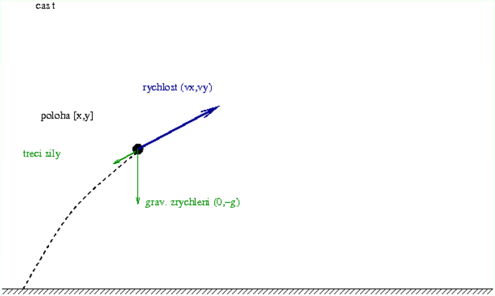
Eulerova metoda

Aplikace na pohyb bodu v gravitacnim poli.

- Jak popsat pohyb hmotného bodu v gravitačním poli?
- Uvažujeme 2D gravitační rovinné pole
- Hmotný bod charakterizován hmotností m, a odporovým součinitelem k.



Pohyb bodu v hmotnem poli



- Pohyb v čase t popsán polohou (x, y) a rychlostí (u, v)
- Gravitační konstanta $g = 9.81 m/s^2$.
- Třecí síla je úměrná kvadrátu rychlosti.

$$
\begin{aligned}
m\ddot{x} &= -k(\dot{x}^2 + \dot{y}^2)^{1/2}\dot{x} \\
m\ddot{y} &= -k(\dot{x}^2 + \dot{y}^2)^{1/2}\dot{y} - mg
\end{aligned}
$$

Vztah k Eulerově metodě.

- Rovnice

$$
\begin{aligned}
m\ddot{x} &= -k(\dot{x}^2 + \dot{y}^2)^{1/2}\dot{x} \\
m\ddot{y} &= -k(\dot{x}^2 + \dot{y}^2)^{1/2}\dot{y} - mg
\end{aligned}
$$

- lze přepsat také jako

$$
\begin{aligned}
\dot{x} &= u \\
\dot{y} &= v \\
m\dot{u} &= -k(u^2 + v^2)^{1/2}u \\
m\dot{v} &= -k(u^2 + v^2)^{1/2}v - mg
\end{aligned}
$$

- neboli

$$\vec{y}' = f(\vec{y}, x)$$

Diskretizace v čase, označení

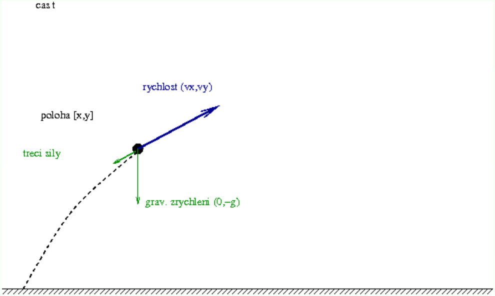
$$t_0 = 0, \qquad t_N = T$$

Ekvidistantní dělení: Δt - krok

$$t_n = a + n\,\Delta t$$

$$\begin{aligned} x^n &\approx x(t_n) \\ y^n &\approx y(t_n) \\ u^n &\approx u(t_n) \\ v^n &\approx v(t_n) \end{aligned}$$

Pohyb bodu v hmotnem poli



Zmena od casu t do casu $t + \Delta t$.

- Poloha

$$\frac{[x^{n+1}, y^{n+1}] - [x^n, y^n]}{\Delta t} = (u^n, v^n).$$

- Rychlost

$$\frac{[u^{n+1}, v^{n+1}] - [u^n, v^n]}{\Delta t} = (0, -9.81) \frac{k|\vec{v}|^n}{m}(u^n, v^n).$$

Program c. 2: Pohyb bodu v gravitacnim poli

```
#include <stdio.h>
#include <math.h>
main()
{
   FILE *fid;
   float x,y,vx,vy;
   float xn,yn,vxn,vyn;
   float pom,float cas,h;
   float uhel,rychlost,k,m;
   /* VSTUPNI UDAJE */
   printf("Uhel a rychlost:");
   scanf("%f%f",&uhel,&rychlost);
   printf("Odpor a hmotnost:");
   scanf("%f%f",&k,&m);
   printf("Casovy krok:");
   scanf("%f",&h);

   cas=0; x=0; y=0;
```

```
    vx=rychlost*cos(uhel*M_PI/180);
    vy=rychlost*sin(uhel*M_PI/180);

    fid=fopen("let.txt","w");
    fprintf(fid,"%f %f %f %f %f \n",cas,x,y,vx,vy);
    while (y>=0)
    {
       xn=x+h*vx;  yn=y+h*vy;
       pom=sqrt(vx*vx+vy*vy);
       vxn=vx+h*(-k/m*pom*vx);
       vyn=vy+h*(-9.81-k/m*pom*vy);
       x=xn;  y=yn;  vx=vxn;  vy=vyn;  cas=cas+h;
       fprintf(fid,"%f %f %f %f %f \n",cas,x,y,vx,vy);
    }
    fclose(fid);
}
```

Sestaveni soustavy rovnic pro y”=f(x), y(a)=A, y(b)=B

```
float A[100][100];
float b[100],x[100];
int sestavmatici(int n)
{
  int i,j;
  for(i=0;i&lt;n;i++)
    for(j=0;j&lt;n;j++)
      A[i][j]=0.0;

  A[0][0]=1;
  for(i=1;i&lt;n-1;i++)
  {
    A[i][i-1]=-1; A[i][i]=2;
    A[i][i+1]=-1;
  }
  A[n-1][n-1]=1;
}
```