**METODA SEČEN pro nalezení kořene nelineární rovnice**

**Kód v jazyce C++:**

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

double f(double x)

{

double y;

y= log((x+1)\*(x+1)) -2 + x;

return y;

}

main()

{ /\* začátek main \*/

int nn, maxi;

float z,y,h,a,b,eps;

/\* c t e n i \*/

cteni:

printf("\n M E T O D A S E C E N\n");

printf(" -----------------------\n\n");

printf(" Zadej levy bod: a = ");

scanf("%f",&a);

printf(" Zadej pravy bod: b = ");

scanf("%f",&b);

printf(" Zadej presnost: eps = ");

scanf("%f",&eps);

printf("\n\n");

/\* o v e r e n i p o d m i n e k \*/

if (b<a) {printf(" a je vetsi nez b\n"); goto cteni;}

if (f(a)\*f(b)>=0) {printf(" f(a)\*f(b) >= 0\n"); goto cteni;}

nn=0; maxi=100;

/\* i t e r a c n i p r o c e s \*/

y=f(a);

navrat:

nn=nn+1;

if (nn>=maxi) {

printf(" pocet iteraci prekrocil povoleny limit\n"); goto konec;}

z=f(b);

h=(b-a)/(z-y)\*z;

a=b; b=b-h;

if (h<0) h=-h; /\* h=abs(h) \*/

printf(" koren( %2d) = %16.10f chyba = %16.10f\n",nn,b,h); /\* t i s k \*/

if (h<eps) goto konec;

y=z;

goto navrat;

konec:

printf("\n");

printf(" koren: %16.10f\n\n",b); /\* v y s l e d e k \*/

system("PAUSE");

} /\* konec main \*/

**Výstup na obrazovku:**

**I T E R A C N I M E T O D A**

**-----------------------------------------**

**Zadej levy bod a: a = 4**

**Zadej pravy bod b: b = 9**

**Zadej presnost: eps = 0.000001**

**koren( 1) = 7.000000**

**koren( 2) = 7.857143 chyba = 0.857143**

**koren( 3) = 7.981818 chyba = 0.124675**

**koren( 4) = 7.997722 chyba = 0.015904**

**koren( 5) = 7.999715 chyba = 0.001993**

**koren( 6) = 7.999964 chyba = 0.000249**

**koren( 7) = 7.999996 chyba = 0.000031**

**koren( 8) = 8.000000 chyba = 0.000004**

**koren( 9) = 8.000000 chyba = 0.000000**

**koren: 8.000000**

**Podmínky konvergence:**

První i druhá derivace funkce musí být spojitá.

Krajní body intervalu < a ; b > musí být „dost blízko“ kořene  [Ralston].

**Iterační formule:**

Volíme  **,** a ostatní počítáme dle:

**Algoritmus:**

1. **Vstup: f(x), x0, x1, **
2. **a=x0 ; b = x1 ; y = f(a) ;**
3. **z = f(b) ;**
4. **h = (b-a) / (z-y) \* z ;**
5. **a = b ; b = b – h ;**
6. **je-li | h | <  jdi na Výstup ;**
7. **Y = z ; jdi na řádek 3 ;**
8. **Výstup: b ;**