Riešenie f(x) = 0

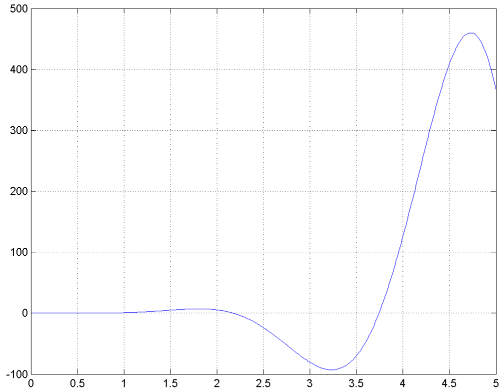
Hľadajme korene funkcie

```
>> f=inline('x.^4.*sin(2*x-1.2)')
```

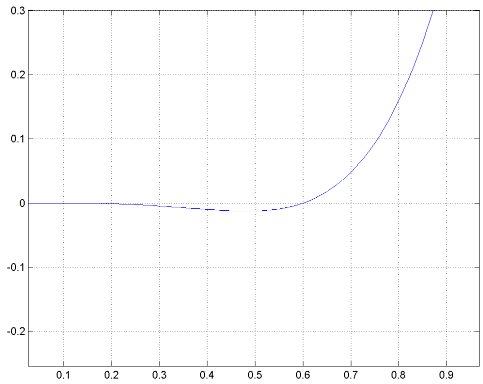
Rýchla úvaha alebo vykreslenie ihneď ukážu, že tých koreňov je nekonečne veľa.
Budeme teda hľadať len prvé dva kladné korene.

```
>> x=0:0.01:5;
>> plot(x,f(x))
>> grid on
```

Netreba sa dať oklamať malým obrázkom grafu



a treba si ho zväčšiť:



Prvé dva korene môžeme už „odseparované“ ohraničiť intervalmi [0.52, 0.67] a [2, 2.5].

Hľadajme prvý z nich **bisekciou**, s presnosťou aspoň 1e-12.

Koľko krokov bude potrebných? Keďže >> 2^(-40) = 9.0949e-013, tak 40 krokov stačí.

```
>> a=0.52; b=0.67; for k=1:40, s=(a+b)/2; if f(s)<0 a=s; else b=s; end, end, s

s =   0.60000000000006
```

Druhý koreň hľadajme **Newtonovou metódou**.
Pozrime sa najprv na derivácie fuknkcie f:

```
>> t=sym('t'); diff(f(t))

ans = 4*t^3*sin(2*t-6/5)+2*t^4*cos(2*t-6/5)
```

```
>> fd=inline('4*x.^3.*sin(2*x-6/5)+2*x.^4.*cos(2*x-6/5)');
>> plot(x,fd(x))
>> grid on

>> t=sym('t'); diff(diff(f(t)))

ans =12*t^2*sin(2*t-6/5)+16*t^3*cos(2*t-6/5)-4*t^4*sin(2*t-6/5)

>> fdd=inline('12*x.^2.*sin(2*x-6/5)+16*x.^3.*cos(2*x-6/5)-4*x.^4.*sin(2*x-6/5)');
>> plot(x,fdd(x))
>> grid on
```

Pohľad na obrázok ukazuje, že f' aj f'' sú záporné na [2, 2.5].

Môžeme bez obáv spustiť Newtonovu metódu:

```
>> s=2;
>> s=s-f(s)/fd(s)
s =    2.27583110057838
>> s=s-f(s)/fd(s)
s =   2.18604673066631
>> s=s-f(s)/fd(s)
s =   2.17120585876330
>> s=s-f(s)/fd(s)
s =   2.17079663545352
>> s=s-f(s)/fd(s)
s =   2.17079632679507
>> s=s-f(s)/fd(s)
s =   2.17079632679490
>> s=s-f(s)/fd(s)
s =   2.17079632679490
```

Ďalej sa už výsledok nemení (vzhľadom na to, čo vie matlab zobraziť). Začali sme zľava, ale preskočilo to doprava a konverguje to sprava. Asi bolo výhodnejšie začať rovno sprava:

```
>> s=2.5;
>> s=s-f(s)/fd(s)
s =   2.26107775868756
>> s=s-f(s)/fd(s)
s =   2.18249170973939
>> s=s-f(s)/fd(s)
s =   2.17103971081212
>> s=s-f(s)/fd(s)
s =   2.17079643586486
>> s=s-f(s)/fd(s)
s =   2.17079632679492
>> s=s-f(s)/fd(s)
s =   2.17079632679490
>> s=s-f(s)/fd(s)
s =   2.17079632679490
```

Ani to nie je oveľa rýchlejšie. Človek by tipol, že keď sa stabilizovalo prvých 14 miest, tak azda presnosť 1e-14 by to mohlo mať. Kontrola:

```
>> eps=1e-14;
>> f(2.17079632679490-eps)*f(2.17079632679490+eps)
ans =   -1.817372345136213e-025
```

Máme znamienkovú zmenu, presnosť teda je dosiahnutá. Pritvrdíme:

```
>> eps=1e-15;
>> f(2.17079632679490-eps)*f(2.17079632679490+eps)
ans =    2.248939093481794e-026
```

Máme smolu, až tak presné to zas nie je...

Úloha:

Nájdite koreň funkcie: f(x) = quad(exp(t.^2-7)-1, 0, x) . Predtým však zvážte, ako túto funkciu opíšete matlabu (daný zápis matlab nevezme...).
Použite obidve metódy a porovnajte ich prednosti a nevýhody.