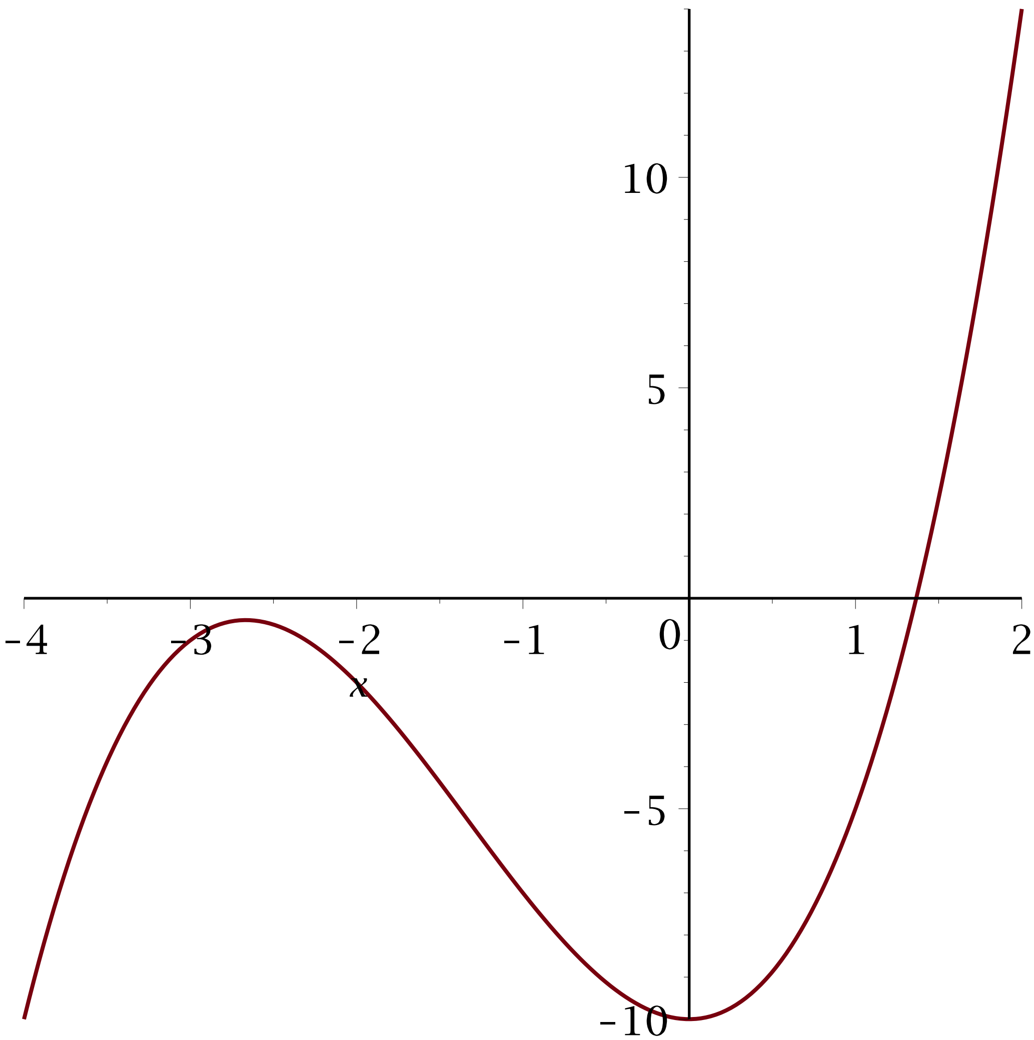
Fixed point iteration for $x^3+4x^2-10=0$, with $g(x)=\left(\frac{10}{x+4}\right)^{\frac{1}{2}}$.

```
> restart;
> Digits:=15;
```

$$Digits := 15 \quad (1)$$

```
> plot(x^3+4*x^2-10,x=-4..2);
```




```
> g:=x->(10/(x+4))^(1/2);
```

$$g := x \to \sqrt{10}\sqrt{\frac{1}{x+4}} \quad (2)$$

```
> p0:=1.5;
```

$$p0 := 1.5 \quad (3)$$

```
> for n from 1 to 100 do
  p[n]:=evalf(g(p0));
  err:=abs(p[n]-p0);
  if err>10^(-10) then
  p0:=p[n];
```

```
else
break
end if;
end do;
```

$$p_1 := 1.34839972492648$$
$$err := 0.15160027507352$$
$$p_2 := 1.36737637199128$$
$$err := 0.01897664706480$$
$$p_3 := 1.36495701540249$$
$$err := 0.00241935658879$$
$$p_4 := 1.36526474811344$$
$$err := 0.00030773271095$$
$$p_5 := 1.36522559416053$$
$$err := 0.00003915395291$$
$$p_6 := 1.36523057567343$$
$$err := 0.00000498151290$$
$$p_7 := 1.36522994187819$$
$$err := 6.3379524\ 10^{-7}$$
$$p_8 := 1.36523002251557$$
$$err := 8.063738\ 10^{-8}$$
$$p_9 := 1.36523001225612$$
$$err := 1.025945\ 10^{-8}$$
$$p_{10} := 1.36523001356143$$
$$err := 1.30531\ 10^{-9}$$
$$p_{11} := 1.36523001339535$$
$$err := 1.6608\ 10^{-10}$$
$$p_{12} := 1.36523001341648$$
$$err := 2.113\ 10^{-11} \quad (4)$$

Aitken's accelerated method

```
> p[0]:=1.5; p[1]:=evalf(g(p[0])); p[2]:=evalf(g(p[1]));q[0]:=evalf
(p[0]-(p[1]-p[0])^2/(p[2]-2*p[1]+p[0]));
```

$$p_0 := 1.5$$
$$p_1 := 1.34839972492648$$
$$p_2 := 1.36737637199128$$
$$q_0 := 1.36526522395726 \quad (5)$$

```
> for n from 3 to 100 do
p[n]:=evalf(g(p[2]));
q[n-2]:=evalf(p[n-2]-(p[n-1]-p[n-2])^2/(p[n]-2*p[n-1]+p[n-2]));
err:=evalf(abs(p[n]-p[2]));
err1:=evalf(abs(q[n-2]-p[2]));
if err1>10^(-10) then
p[2]:=p[n];
else
```

```
break
end if;
end do;
```

$$p_3 := 1.36495701540249$$
$$q_1 := 1.36523058454178$$
$$err := 0.00241935658879$$
$$err1 := 0.00214578744950$$
$$p_4 := 1.36526474811344$$
$$q_2 := 1.36495701540249$$
$$err := 0.00030773271095$$
$$err1 := 0. \quad (6)$$

```
> fsolve({g(x)=x},{x});
```

$$\{x = 1.36523001341410\} \quad (7)$$

Aitken-Steffensen's accelerated method

```
> g:=x->(10/(x+4))^(1/2);
```

$$g := x \to \sqrt{10}\,\sqrt{\frac{1}{x+4}} \quad (8)$$

```
> p0:=1.5;
```

$$p0 := 1.5 \quad (9)$$

```
> for n from 1 to 10 do
p1:=evalf(g(p0));
p2:=evalf(g(p1));
q[n]:=evalf(p0-(p1-p0)^2/(p2-2*p1+p0));
err:=evalf(abs(q[n]-p0));
if err>10^(-10) then
p0:=q[n]
else
break
end if
end do;
```

$$p1 := 1.34839972492648$$
$$p2 := 1.36737637199128$$
$$q_1 := 1.36526522395726$$
$$err := 0.13473477604274$$
$$p1 := 1.36522553361979$$
$$p2 := 1.36523058337601$$
$$q_2 := 1.36523001341659$$
$$err := 0.00003521054067$$
$$p1 := 1.36523001341378$$
$$p2 := 1.36523001341414$$
$$q_3 := 1.36523001341410$$
$$err := 2.49\ 10^{-12} \quad (10)$$

```
> fsolve({g(x)=x},{x});
```

(11)

$$\{x = 1.36523001341410\}$$ **(11)**

```
> Digits:=15;
```

$$Digits := 15$$ **(12)**

Another example

```
> Digits:=10;
```

$$Digits := 10$$ **(13)**

```
> p:=n->sum((-1)^(k-1)/k,k=1..n);
```

$$p := n \to \sum_{k=1}^{n} \frac{(-1)^{k-1}}{k}$$ **(14)**

```
> sum((-1)^(k-1)/k,k=1..infinity);
```

$$\ln(2)$$ **(15)**

```
> evalf(sum((-1)^(k-1)/k,k=1..100)-ln(2));
```

$$-0.0049750013$$ **(16)**

```
> q:=n->p(n)-(p(n+1)-p(n))^2/(p(n+2)-2*p(n+1)+p(n));
```

$$q := n \to p(n) - \frac{(p(n+1) - p(n))^2}{p(n+2) - 2\,p(n+1) + p(n)}$$ **(17)**

```
> for n from 2 to 15 do
  s[n]:=evalf(abs(p(n)-ln(2))):
  aitken[n]:=evalf(abs(q(n)-ln(2)))
  end do;
```

$$s_2 := 0.1931471806$$
$$aitken_2 := 0.0026709901$$
$$s_3 := 0.1401861527$$
$$aitken_3 := 0.0012972638$$
$$s_4 := 0.1098138473$$
$$aitken_4 := 0.0007229382$$
$$s_5 := 0.0901861527$$
$$aitken_5 := 0.0004425630$$
$$s_6 := 0.0764805139$$
$$aitken_6 := 0.0002900377$$
$$s_7 := 0.0663766289$$
$$aitken_7 := 0.0002001583$$
$$s_8 := 0.0586233711$$
$$aitken_8 := 0.0001438389$$
$$s_9 := 0.0524877400$$
$$aitken_9 := 0.0001067877$$
$$s_{10} := 0.0475122600$$
$$aitken_{10} := 0.0000814299$$
$$s_{11} := 0.0433968309$$

$$aitken_{11} := 0.0000634976$$
$$s_{12} := 0.0399365024$$
$$aitken_{12} := 0.0000504625$$
$$s_{13} := 0.0369865745$$
$$aitken_{13} := 0.0000407617$$
$$s_{14} := 0.0344419969$$
$$aitken_{14} := 0.0000333947$$
$$s_{15} := 0.0322246698$$
$$aitken_{15} := 0.0000277001 \quad (18)$$

```
> evalf(abs(p(3)-ln(2)));
```

0.1401861527 **(19)**

```
> evalf(abs(q(3)-ln(2)));
```

0.0012972638 **(20)**

```
> restart;
```

Steffensen's method applied to fixed point iteration

```
> Digits:=15;
```

$$Digits := 15 \quad (21)$$

```
> g:=x->(10/(x+4))^(1/2);
```

$$g := x \rightarrow \sqrt{10}\,\sqrt{\frac{1}{x+4}} \quad (22)$$

```
> #G:=x->(x*g(g(x))-(g(x))^2)/(g(g(x))-2*g(x)+x);
  G:=x->x-(g(x)-x)^2/(g(g(x))-2*g(x)+x);
```

$$G := x \rightarrow x - \frac{(g(x) - x)^2}{g(g(x)) - 2\,g(x) + x} \quad (23)$$

```
> p0:=1.0;
```

$$p0 := 1.0 \quad (24)$$

```
> for n from 1 to 100 do
  p[n]:=evalf(G(p0));
  err:=abs(p[n]-p0);
  if err>10^(-10) then
  p0:=p[n];
  else
  break
  end if;
  end do;
```

$$p_1 := 1.36552548138721$$
$$err := 0.36552548138721$$
$$p_2 := 1.36523001358933$$
$$err := 0.00029546779788$$
$$p_3 := 1.36523001341410$$
$$err := 1.7523\ 10^{-10}$$

$$p_4 := 1.36523001341410$$
$$err := 0. \tag{25}$$

Steffensen's method combined with Newton's method

Newton's method

```
> Digits:=100;
```
$$Digits := 100 \tag{26}$$

```
> f:=x->x^3+4*x^2-10;
```
$$f := x \rightarrow x^3 + 4\,x^2 - 10 \tag{27}$$

```
> fp:=D(f);
```
$$fp := x \rightarrow 3\,x^2 + 8\,x \tag{28}$$

```
> p0:=1.5;
```
$$p0 := 1.5 \tag{29}$$

```
> for n from 1 to 20 do
  p[n]:=evalf(p0-f(p0)/fp(p0));
  err:=abs(p[n]-p0);
  if err>=10^(-96) then
  p0:=p[n]
  else
  break
  end if
  end do;
```

$p_1 :=$
1.3733333333333333333333333333333333333333333333333333333333\
33333333333333333333333333333333333333333

$err :=$
0.1266666666666666666666666666666666666666666666666666666666\
66666666666666666666666666666666666666667

$p_2 :=$
1.3652620148746266212381755974950245833644853158455004076459\
42587786286719286815412932879048422641617

$err :=$
0.0080713184587067120951577358383087499688480174878329256873\
90745547046614046517920400454284910691716

$p_3 :=$
1.3652300139161466492910961528563044690908152151648048029545\
34183821212745811822506811491951603025372

$err :=$
0.0000320009584799719470794446387201142736701006806956046914\
08403965073973474992906121387096819616245

$p_4 :=$
1.3652300134140968458843762461302611999366553554143956242256\
06485688546788632488402327120488246964030

err :=
5.020498034067199067260432691541598597504091787289276981 3266\
5957179334104484371463356061342 10^{-10}

p_5 :=
1.3652300134140968457608068289816660783386505669739784001034\
04633291949352843791255215550044903677088

err :=
1.2356941714859512159800478844041722412220185239659743578869\
7147111570443343286942 10^{-19}

p_6 :=
1.3652300134140968457608068289816660783311647467712650718237\
87354745502933196112029690934391982943952

err :=
7.4858202027133282796172785464464196476792255246156529207331\
36 10^{-39}

p_7 :=
1.3652300134140968457608068289816660783311647467712650718237\
87354745502933196084557317633355389556551

err := 2.7472373301036593387401 10^{-77}

p_8 :=
1.3652300134140968457608068289816660783311647467712650718237\
87354745502933196084557317633355389556551

err := 0. **(30)**

Steffensen's combined with Newton's method

```
> Digits:=100;
```

Digits := 100 **(31)**

```
> p0:=1.5;
```

p0 := 1.5 **(32)**

```
> for n from 1 to 10 do
  p1:=p0-f(p0)/fp(p0);
  p2:=p1-f(p1)/fp(p1);
  q[n]:=evalf(p0-(p1-p0)^2/(p2-2*p1+p0));
  err:=evalf(abs(q[n]-p0));
  if err>=10^(-96) then
  p0:=q[n]
  else
  break
  end if
  end do;
```

p1 :=
1.37333333333333333333333333333333333333333333333333333333333\
33333333333333333333333333333333333333333

p2 :=
1.3652620148746266212381755974950245833644853158455004076459\
42587786286719286815412932879048422641617

$q_1 :=$
1.36471270006046018183684876969233752752079031724484092127626\
80215831441576071457090382268055349585335
$err :=$
0.13528729993953981816315123030766247247920968275515907872370\
19784168558423928542909617731944650414665
$p1 :=$
1.36523014466115799347229885824339218343060483530069132008851\
10231358284756414107465755623032121115678
$p2 :=$
1.36523001341410529070003791521287836939678731782896631948244\
49107875927115557863004152264129012101426
$q_2 :=$
1.36523001344738696123785554678960832225827435223667206307458\
88094532980866479788240567169610150617498
$err :=$
0.00051731338692677940100677709727079473748403499183114179833\
07878701539290408331150184901554801032163
$p1 :=$
1.36523001341409684576135013935712460820949254616189835837833\
57885232221408770717610007010186652857294
$p2 :=$
1.36523001341409684576080682898166607833116489148621307157611\
9143154048073192489323401707735398412737 0
$q_3 :=$
1.36523001341409684576080682898165721124975467424777115890830\
04297102156726195561662959603825364403379
$err :=$
3.32901154770487178079511110085196779889009041662837974308241\
40284226577607565784786214119 10^{-11}
$p1 :=$
1.36523001341409684576080682898166607833116474677126507182378\
87354784048886155419826681571052785325000
$p2 :=$
1.36523001341409684576080682898166607833116474677126507182378\
87354745502933196084557317633355389556552
$q_4 :=$
1.36523001341409684576080682898166607833116474677126507182378\
87354745502933196084557317633355389556720
$err :=$
8.86708141007252349391291548305764334620700052289435802953000\
25153341 10^{-33}
$p1 :=$
1.36523001341409684576080682898166607833116474677126507182378\
87354745502933196084557317633355389556552
$p2 :=$
1.36523001341409684576080682898166607833116474677126507182378\

87354745502933196084557317633355389556551

$q_5 :=$

1.3652300134140968457608068289816660783311647467712650718237\
87354745502933196084557317633355389556551

$$err := 1.69\ 10^{-97} \tag{33}$$