###
Section 0.4
****
Problem 5
5. Consider a right triangle whose legs are of length 3344556600 and 1.2222222. How much longer is the hypotenuse than the longer leg? Give your answer with at least four correct digits.

sqrt(*largeNumber*^2 + *SmallNumber*^2) - *largeNumber* =
(sqrt(*largeNumber*^2 + *SmallNumber*^2) - *largeNumber*) *
(sqrt(*largeNumber*^2 + *SmallNumber*^2) + *largeNumber*) /
(sqrt(*largeNumber*^2 + *SmallNumber*^2) + *largeNumber*) =
(*largeNumber*^2 + *SmallNumber*^2 - *largeNumber*^2) /
(sqrt(*largeNumber*^2 + *SmallNumber*^2) + *largeNumber*) =
(*SmallNumber*^2) / (sqrt(*largeNumber*^2 + *SmallNumber*^2) + *largeNumber*)

formula for difference =
(*SmallNumber*^2) / (sqrt(*largeNumber*^2 + *SmallNumber*^2) + *large*Number)

$$false$$

$$\frac{1.493827106}{\sqrt{largeNumber^2 + 1.493827106} + large\ Number} \tag{1}$$

```
> LargeNumber := 3344556600
```

$$LargeNumber := 3344556600 \tag{2}$$

```
> SmallNumber := 1.2222222
```

$$SmallNumber := 1.2222222 \tag{3}$$

```
> NumberOfSignificantDigits := 4
```

$$NumberOfSignificantDigits := 4 \tag{4}$$

```
> difference := evalf[NumberOfSignificantDigits](SmallNumber^2/((LargeNumber^2
      + SmallNumber^2)^.5 + LargeNumber))
```

$$difference := 2.232\ 10^{-10} \tag{5}$$

Answer: *difference* $:= 2.232\ 10^{-10}$
###
Section 1.2
****

$$difference := 2.232000000\ 10^{-10} \tag{6}$$

```
fixedpoint:=proc(g,X0,TOL,N)
          local i,x0;x0:=X0;
          i:=1;
          while i<=N do
               r:=g(x0);
               printf("Iteration %d: %.8g\n",i,r);
               if abs(r-x0)<TOL then
                  printf("Fixed point r=%.8g, g(r)=%.8g\n",r,g(r));
                  printf("Number of iterations needed: %d",i);return
();
               break;
```

```
            end if;
            i:=i+1;x0:=r;
        end do;
        printf("The method failed after %d iterations.\n",N);
        printf("r=%.8g, g(r)=%.8g\n",r,g(r));
        return();
    end proc:
```

Problem 1a
1. Apply Fixed-Point Iteration to find the solution of each equation to eight correct decimal places. (a) **x3 = 2x + 2**
Acording to Newton Method
Lets isolate the x^3 and take the cubic root
x^3=2*x+2 => g(x) = (2*x+2)^(1/3)

> $g(x) := x \rightarrow (2*x+2)\hat{}(1/3);$

$$g := x \rightarrow x \rightarrow (2*x+2)\hat{}(1/3) \quad \textbf{(7)}$$

> $fixedpoint\left(g(x), 0.5, 0.5 \cdot 10^{-8}, 50\right);$

```
Iteration 1: 1.4422496
Iteration 2: 1.6967063
Iteration 3: 1.753697
Iteration 4: 1.7659648
Iteration 5: 1.7685834
Iteration 6: 1.7691414
Iteration 7: 1.7692602
Iteration 8: 1.7692855
Iteration 9: 1.7692909
Iteration 10: 1.769292
Iteration 11: 1.7692923
Iteration 12: 1.7692923
Iteration 13: 1.7692924
Iteration 14: 1.7692924
Fixed point r=1.7692924, g(r)=1.7692924
Number of iterations needed: 14
```

Answer : root= 1.7692924, Number Of Iterations 14

Problem 1c
1. Apply Fixed-Point Iteration to find the solution of each equation to eight correct decimal places. (c) e^x + sin x = 4.
Lets isolate x
e^x = 4-sinx
x=ln(4-sinx) => g(x)= ln(4-sinx)

> $g(x) := x \rightarrow \ln(4 - \sin(x));$

$$g := x \rightarrow x \rightarrow \ln(4 - \sin(x)) \quad \textbf{(8)}$$

> $fixedpoint\left(g(x), 0.5, 0.5 \cdot 10^{-8}, 50\right);$

```
Iteration 1: 1.2586242
Iteration 2: 1.1145943
Iteration 3: 1.1321334
```

```
Iteration 4: 1.1296844
Iteration 5: 1.1300213
Iteration 6: 1.1299749
Iteration 7: 1.1299813
Iteration 8: 1.1299804
Iteration 9: 1.1299805
Iteration 10: 1.1299805
Iteration 11: 1.1299805
Fixed point r=1.1299805, g(r)=1.1299805
Number of iterations needed: 11
```

Answer : root= 1.1299805, Number Of Iterations 11

Problem 4

4. Calculate the cube roots of the following numbers to eight correct decimal places, by using Fixed-Point Iteration with g(x) = (2x + A/x2)/3, where A is (a) 2 (b) 3 (c) 5. State your initial guess and the number of steps needed.

Guess:
evaluate g(x) = (2x + A/x2)/3 at a^(1/3)

g(A^1/3) = 0
g'(x) = 1/3(2+(-2)*A*x^(-3)) = 2/3(1-A/x^3)
|g'(A^1/3)| = | 2/3(1- A/(A^1/3)^3| = **0** < 1 =>
|g''(A^1/3)| != 0
m = 2
lim(en+1/en) = (m-1/m) =(2-1)/2 = 1/2 => en+1 = (1/2)*en
FPI is locally convergent to A^1/3 with rate of convergence S=0 =>
at least quadratic convergence
which means that
(1/2)^n < 0.5 * 10^8
n-1 >=8log2(10)
n > 1+ 8*log2(10) = 27.57542
Expected Number of Steps (needed) to eight correct decimal places = 28

a) 2
Initial Guess : 1.0
Expected Number Of Steps = 28

> $A := 2$

$$A := 2 \tag{9}$$

> $fun := x \rightarrow \frac{1}{2} \cdot \left(x + \frac{A}{x^2} \right)$

$$fun := x \mapsto \frac{x}{2} + \frac{A}{2\,x^2} \tag{10}$$

> $fixedpoint(fun, 1.0, 0.5 \cdot 10^{-8}, 50);$

```
Iteration 1: 1.5
Iteration 2: 1.1944444
Iteration 3: 1.2981416
Iteration 4: 1.2424822
Iteration 5: 1.2690094
Iteration 6: 1.2554743
Iteration 7: 1.2621681
Iteration 8: 1.2588035
Iteration 9: 1.2604813
```

```
Iteration 10: 1.2596413
Iteration 11: 1.260061
Iteration 12: 1.2598511
Iteration 13: 1.259956
Iteration 14: 1.2599036
Iteration 15: 1.2599298
Iteration 16: 1.2599167
Iteration 17: 1.2599232
Iteration 18: 1.25992
Iteration 19: 1.2599216
Iteration 20: 1.2599208
Iteration 21: 1.2599212
Iteration 22: 1.259921
Iteration 23: 1.2599211
Iteration 24: 1.259921
Iteration 25: 1.2599211
Iteration 26: 1.259921
Iteration 27: 1.2599211
Iteration 28: 1.259921
Fixed point r=1.259921, g(r)=1.259921
Number of iterations needed: 28
```

Answer: FPI root = 1.259921, Number Of steps to eight correct decimal places = 28

b) **3**

Initial guess : 1.0

Expected Number Of Steps = 28

> $A := 3$

$$A := 3 \tag{11}$$

> $fun := x \rightarrow \frac{1}{2} \cdot \left(x + \frac{A}{x^2}\right)$

$$fun := x \mapsto \frac{x}{2} + \frac{A}{2\,x^2} \tag{12}$$

> $fixedpoint(fun, 1.0, 0.5 \cdot 10^{-8}, 50)$

```
Iteration 1: 2
Iteration 2: 1.375
Iteration 3: 1.4808884
Iteration 4: 1.4244292
Iteration 5: 1.4514956
Iteration 6: 1.4377147
Iteration 7: 1.4445385
Iteration 8: 1.4411106
Iteration 9: 1.4428204
Iteration 10: 1.4419645
Iteration 11: 1.4423922
Iteration 12: 1.4421783
Iteration 13: 1.4422852
Iteration 14: 1.4422317
Iteration 15: 1.4422585
Iteration 16: 1.4422451
Iteration 17: 1.4422518
Iteration 18: 1.4422485
Iteration 19: 1.4422501
Iteration 20: 1.4422493
```

```
Iteration 21: 1.4422497
Iteration 22: 1.4422495
Iteration 23: 1.4422496
Iteration 24: 1.4422496
Iteration 25: 1.4422496
Iteration 26: 1.4422496
Iteration 27: 1.4422496
Iteration 28: 1.4422496
Fixed point r=1.4422496, g(r)=1.4422496
Number of iterations needed: 28
```

Answer: FPI root = 1.4422496, Number Of steps to eight correct decimal places = 28

c) 5
Initial guess 1.0
Expected Number Of Steps = 28

> $A := 5$

$$A := 5 \quad (13)$$

> $fun := x \rightarrow \frac{1}{2} \cdot \left(x + \frac{A}{x^2}\right)$

$$fun := x \mapsto \frac{x}{2} + \frac{A}{2\,x^2} \quad (14)$$

> $fixedpoint(fun, 0.5, 0.5 \cdot 10^{-8}, 50)$

```
Iteration 1: 10.25
Iteration 2: 5.1487954
Iteration 3: 2.6687014
Iteration 4: 1.6853773
Iteration 5: 1.7228164
Iteration 6: 1.7036989
Iteration 7: 1.7131492
Iteration 8: 1.7083981
Iteration 9: 1.710767
Iteration 10: 1.7095809
Iteration 11: 1.7101736
Iteration 12: 1.7098772
Iteration 13: 1.7100253
Iteration 14: 1.7099512
Iteration 15: 1.7099883
Iteration 16: 1.7099698
Iteration 17: 1.709979
Iteration 18: 1.7099744
Iteration 19: 1.7099767
Iteration 20: 1.7099756
Iteration 21: 1.7099761
Iteration 22: 1.7099758
Iteration 23: 1.709976
Iteration 24: 1.7099759
Iteration 25: 1.709976
Iteration 26: 1.7099759
Iteration 27: 1.709976
Iteration 28: 1.7099759
Iteration 29: 1.7099759
Fixed point r=1.7099759, g(r)=1.7099759
Number of iterations needed: 29
```

Answer: FPI root = 1.7099759, Number Of steps to eight correct decimal places = 29

A = 25
Initial guess : 3.0
Expected Number Of Steps = 28

> $A := 25$

$$A := 25 \tag{15}$$

> $fun := x \rightarrow \frac{1}{2} \cdot \left(x + \frac{A}{x^2}\right)$

$$fun := x \mapsto \frac{x}{2} + \frac{A}{2\,x^2} \tag{16}$$

> $fixedpoint(fun, 3.0, 0.5 \cdot 10^{-8}, 50)$

```
Iteration 1: 2.8888889
Iteration 2: 2.9422255
Iteration 3: 2.9150825
Iteration 4: 2.9285265
Iteration 5: 2.9217738
Iteration 6: 2.9251423
Iteration 7: 2.9234561
Iteration 8: 2.9242987
Iteration 9: 2.9238773
Iteration 10: 2.924088
Iteration 11: 2.9239826
Iteration 12: 2.9240353
Iteration 13: 2.924009
Iteration 14: 2.9240221
Iteration 15: 2.9240155
Iteration 16: 2.9240188
Iteration 17: 2.9240172
Iteration 18: 2.924018
Iteration 19: 2.9240176
Iteration 20: 2.9240178
Iteration 21: 2.9240177
Iteration 22: 2.9240178
Iteration 23: 2.9240177
Iteration 24: 2.9240177
Iteration 25: 2.9240177
Iteration 26: 2.9240177
Fixed point r=2.9240177, g(r)=2.9240177
Number of iterations needed: 26
```

Answer: FPI root = 2.9240177, Number Of steps to eight correct decimal places = 26

A = 30
Initial guess: 3.0
Expected Number Of Steps = 28

> $A := 30$

$$A := 30 \tag{17}$$

> $fun := x \rightarrow \frac{1}{2} \cdot \left(x + \frac{A}{x^2}\right)$

(18)

$$fun := x \mapsto \frac{x}{2} + \frac{A}{2\,x^2} \tag{18}$$

> $fixedpoint\left(fun, 3.0, 0.5 \cdot 10^{-8}, 50\right)$

```
Iteration 1: 3.1666667
Iteration 2: 3.0791782
Iteration 3: 3.1216442
Iteration 4: 3.1001263
Iteration 5: 3.1108101
Iteration 6: 3.1054499
Iteration 7: 3.1081253
Iteration 8: 3.1067865
Iteration 9: 3.1074556
Iteration 10: 3.107121
Iteration 11: 3.1072883
Iteration 12: 3.1072046
Iteration 13: 3.1072464
Iteration 14: 3.1072255
Iteration 15: 3.107236
Iteration 16: 3.1072308
Iteration 17: 3.1072334
Iteration 18: 3.1072321
Iteration 19: 3.1072327
Iteration 20: 3.1072324
Iteration 21: 3.1072326
Iteration 22: 3.1072325
Iteration 23: 3.1072325
Iteration 24: 3.1072325
Iteration 25: 3.1072325
Iteration 26: 3.1072325
Iteration 27: 3.1072325
Fixed point r=3.1072325, g(r)=3.1072325
Number of iterations needed: 27
```

Answer: FPI root = 2.9240177, Number Of steps to eight correct decimal places = 27

Problem: In the 1991 Gulf War, the Patriot missile defense system failed due to roundoff error. The troubles stemmed from a computer that performed the tracking calculations with an internal
clock whose integer values in tenths of a second were converted to seconds by multiplying by a 24-bit binary approximation to one tenth:
0.1(10) ≈ 0.000110011001100110011100(2)
Call it x.

> $actual\_dec\_number := 0.1$

$$actual\_dec\_number := 0.1 \tag{19}$$

> $y := 0.000110011001100110011100$

$$y := 0.000110011001100110011100 \tag{20}$$

(a) Convert the binary number to a decimal.

> $x := convert(y, decimal, binary)$

$$x := 0.09999990463 \tag{21}$$

(b) What is the absolute error in this number; i.e., what is the absolute value of the difference between x and 0.1 ?

> $absolute\_error := evalf\left(\text{abs}(actual\_dec\_number - x)\right)$

$$absolute\_error := 9.537\; 10^{-8} \tag{22}$$

(c) What is the time error in seconds after 100 hours of operation (i.e., |3,600,000(0.1-x)|)?

> $time\_error := \mathrm{abs}(3600000 \cdot absolute\_error)$

$$time\_error := 0.3433320000 \tag{23}$$

(d) During the 1991 war, a Scud missile traveled at approximately MACH 5 (3750 miles per hour). Find the distance that a Scud missile would travel during the time error computed in (c).

> $speed := \frac{3750 \cdot 1.60934 \cdot 1000}{3600}$

$$speed := 1676.395833 \tag{24}$$

>

> $distance := speed \cdot time\_error$

$$distance := 575.5603341 \tag{25}$$

Answer: The distance that a Scud missile would travel during the time error computed in (c):
distance =

575.56 meters or
0.5722/1000 = 0.57556 km or
0.0005722 /1.6 = 0.357625 miles

>