Scilab Textbook Companion for Surveying Volume 3 by B. C. Punmia, A. K. Jain And A. K. Jain[1]

Created by
Avik Kumar Das
Geodesy
Civil Engineering
IIT, Bombay
College Teacher
Prof Raaj Ramshankaran
Cross-Checked by
Mukul and Lavitha

October 11, 2013

[1]Funded by a grant from the National Mission on Education through ICT, http://spoken-tutorial.org/NMEICT-Intro. This Textbook Companion and Scilab codes written in it can be downloaded from the "Textbook Companion Project" section at the website http://scilab.in

Book Description

Title: Surveying Volume 3

Author: B. C. Punmia, A. K. Jain And A. K. Jain

Publisher: Laxmi Publication, New Delhi

Edition: 9

Year: 1990

ISBN: 81-7008-054-18

Scilab numbering policy used in this document and the relation to the above book.

Exa Example (Solved example)

Eqn Equation (Particular equation of the above book)

AP Appendix to Example(Scilab Code that is an Appednix to a particular Example of the above book)

For example, Exa 3.51 means solved example 3.51 of this book. Sec 2.3 means a scilab code whose theory is explained in Section 2.3 of the book.

Contents

List of Scilab Codes

Chapter 1

Field Astronomy

Scilab code Exa 1.1.1 astronomical triangle

```
1
2
3 //exapple 1.1.1
4 clc; funcprot(0);
5 // Initialization of Variable
6 a=40;//longitude of A
7 b=73;//longitude of B
8 dol=b-a;//difference of longitude
9 disp("difference of longitude is");
10 a=modulo(dol*3600,60);
11 printf("seconds %.2f",a);
12 b=modulo(dol*3600-a,3600)/60;
13 printf("  minutes %i",b);
14 c=(dol*3600-b*60-a)/3600;
15 printf("  degrees %i",c);
```

Scilab code Exa 1.1.2 astronomical triangle

```
1
2
3 //exapple 1.1.2
4 clc; funcprot(0);
5 // Initialization of Variable
6 a=20;//longitude of A
7 b=150;//longitude of B
8 dol=b-a;//difference of longitude
9 disp("difference of longitude is");
10 a=modulo(dol*3600,60);
11 printf("seconds %.2f",a);
12 b=modulo(dol*3600-a,3600)/60;
13 printf("  minutes %i",b);
14 c=(dol*3600-b*60-a)/3600;
15 printf("  degrees %i",c);
```

Scilab code Exa 1.1.3 astronomical triangle

```
1
2
3 //exapple 1.1.3
4 clc; funcprot(0);
5 // Initialization of Variable
6 a=-20;//longitude of A
7 b=50;//longitude of B
8 dol=b-a;//difference of longitude
9 disp("difference of longitude is");
10 a=modulo(dol*3600,60);
11 printf("seconds %.2f",a);
12 b=modulo(dol*3600-a,3600)/60;
13 printf("  minutes %i",b);
14 c=(dol*3600-b*60-a)/3600;
15 printf("  degrees %i",c);
```

Scilab code Exa 1.1.4 astronomical triangle

```
1
2
3 //exapple 1.1.4
4 clc; funcprot(0);
5 // Initialization of Variable
6 a=-40;//longitude of A
7 b=150;//longitude of B
8 dol=b-a;//difference of longitude
9 if  dol>180 then
10     dol=360-dol;
11     disp("difference of longitude is");
12 a=modulo(dol*3600,60);
13 printf("seconds %.2f",a);
14 b=modulo(dol*3600-a,3600)/60;
15 printf("  minutes %i",b);
16 c=(dol*3600-b*60-a)/3600;
17 printf("  degrees %i",c);
18 end
```

Scilab code Exa 1.2.1 astronomical triangle

```
1
2
3 //exapple 1.2.1
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14159;
7 latA=28+42/60;//latitude of A
8 lonA=31*60+12;//longitude of A
9 latB=28+42/60;//latitude of B
```

```
10 lonB=47*60+24;//longitude of B
11 d=(lonB-lonA)*cos(latA/180*pi);
12 disp(d*1.852,"distance between A & B in (km):");
```

Scilab code Exa 1.2.2 astronomical triangle

```
1
2
3 //exapple 1.2.2
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14259;
7 latA=12+36/60;//latitude of A
8 lonA=-115*60-6;//longitude of A
9 latB=12+36/60;//latitude of B
10 lonB=150*60+24;//longitude of B
11 d=(360*60-lonB+lonA)*cos(latA/180*pi);
12 disp(d*1.852,"distance between A & B in (km):");
```

Scilab code Exa 1.3 astronomical triangle

```
1
2
3 //exapple 1.3
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14159
7 latA=15;
8 latB=12+6/60;
9 lonA=50+12/60;
10 lonB=54;
11 Re=6370;//radius of earth
12 b=(90-latA)*pi/180;;
```

```
13 a=(90-latB)*pi/180;
14 P=(lonB-lonA)*pi/180;
15 p=acos(cos(P)*sin(a)*sin(b)+cos(a)*cos(b));//
      spehericaltriangle law
16 disp(p*Re,"distance from A to B in (km):");
17 x=atan(cos(a/2-b/2)/cos(a/2+b/2)*tan(pi/2-P/2));//x=
      A/2+B/2 //spehericaltriangle law
18 y=atan(sin(a/2-b/2)/sin(a/2+b/2)*tan(pi/2-P/2));//x=
      A/2-B/2 //spehericaltriangle law
19 dol=pi-x-y;
20 disp("direction of B from A  towards east of south:"
      );
21 a=modulo(dol*3600,60);
22 printf("seconds %.2f",a);
23 b=modulo(dol*3600-a,3600)/60;
24 printf("  minutes %i",b);
25 c=(dol*3600-b*60-a)/3600;
26 printf("  degrees %i",c);
```

Scilab code Exa 1.4 astronomical triangle

```
1
2
3 //exapple 1.4
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14159;
7 latA=45;
8 p=(300/60)*pi/180;//side AB
9 b=(90-latA)*pi/180;//side PA
10 //calculation
11 a=acos(cos(p)*cos(b));//side BP
12 BC=a*180/pi-latA;
13 d=BC*1.852*60;
14 disp(d,"disatance of BC in (km):")
```

Scilab code Exa 1.5 zenith distance

```
1
2
3 //exapple 1.5
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14159;
7 delta=42+15/60;//declination of star
8 theta=26+40/60;//latitude of star
9 zend=90-theta-90+delta;
10 alt=90-zend;
11 disp("zenith distace of star in (degrees):");
12 a=modulo(zend*3600,60);
13 printf("seconds %.2f",a);
14 b=modulo(zend*3600-a,3600)/60;
15 printf("  minutes %i",b);
16 c=(zend*3600-b*60-a)/3600;
17 printf("  degrees %i",c);
18 disp(,"altitude of star in (degrees):");
19 a=modulo(alt*3600,60);
20 printf("seconds %.2f",a);
21 b=modulo(alt*3600-a,3600)/60;
22 printf("  minutes %i",b);
23 c=(alt*3600-b*60-a)/3600;
24 printf("  degrees %i",c);
```

Scilab code Exa 1.6.1 zenith distance and altitude of star

```
1
2
```

```
3  //exapple 1.6.1
4  clc; funcprot(0);
5  // Initialization of Variable
6  pi=3.14159;
7  delta=23+20/60;//declination of star
8  theta=26+40/60;//latitude of sta
9  zend=90+theta-90-delta;
10 alt=90-zend;
11 disp("zenith distace of star in (degrees):");
12 a=modulo(zend*3600,60);
13 printf("seconds %.2f",a);
14 b=modulo(zend*3600-a,3600)/60;
15 printf("  minutes %i",b);
16 c=(zend*3600-b*60-a)/3600;
17 printf("  degrees %i",c);
18 disp(,"altitude of star in (degrees):");
19 a=modulo(alt*3600,60);
20 printf("seconds %.2f",a);
21 b=modulo(alt*3600-a,3600)/60;
22 printf("  minutes %i",b);
23 c=(alt*3600-b*60-a)/3600;
24 printf("  degrees %i",c);
```

Scilab code Exa 1.6.2 zenith distance and altitude at upper culmination

```
1
2
3  //exapple 1.6.2
4  clc; funcprot(0);
5  // Initialization of Variable
6  pi=3.14159;
7  delta=65+40/60;//declination of star
8  theta=26+40/60;//latitude of star
9  zend=90-theta-90+delta;
10 alt=90-zend;
```

```
11 disp("zenith distace of star in (degrees):");
12 a=modulo(zend*3600,60);
13 printf("seconds %.2f",a);
14 b=modulo(zend*3600-a,3600)/60;
15 printf("  minutes %i",b);
16 c=(zend*3600-b*60-a)/3600;
17 printf("  degrees %i",c);
18 disp(,"altitude of star in (degrees):");
19 a=modulo(alt*3600,60);
20 printf("seconds %.2f",a);
21 b=modulo(alt*3600-a,3600)/60;
22 printf("  minutes %i",b);
23 c=(alt*3600-b*60-a)/3600;
24 printf("  degrees %i",c);
```

Scilab code Exa 1.7 astronomical triangle

```
1
2
3 //exapple 1.7
4 clc; funcprot(0);
5 // Initialization of Variable
6
7 pi=3.14159;
8 delta=85+20/60;
9 theta=46+50/60;
10 if delta>90-theta then
11     zend=180-theta-delta;
12     disp("zenith distance in (degrees):");
13     a=modulo(zend*3600,60);
14 printf("seconds %.2f",a);
15 b=modulo(zend*3600-a,3600)/60;
16 printf("  minutes %i",b);
17 c=(zend*3600-b*60-a)/3600;
18 printf("  degrees %i",c);
```

```
19     zend1=90-zend;
20      disp("altitude of star in (degrees):");
21      a=modulo(zend1*3600,60);
22 printf("seconds %.2f",a);
23 b=modulo(zend1*3600-a,3600)/60;
24 printf("  minutes %i",b);
25 c=(zend1*3600-b*60-a)/3600;
26 printf("  degrees %i",c);
27 end
```

Scilab code Exa 1.8 astronomical triangle

```
1
2
3 //exapple 1.8
4 clc; funcprot(0);
5 // Initialization of Variable
6
7 pi=3.14159;
8 delta=56+10/60;
9 theta=56+10/60;
10 if delta>90-theta then
11     zend=180-theta-delta;
12      disp("zenith distance in (degrees):");
13     a=modulo(zend*3600,60);
14 printf("seconds %.2f",a);
15 b=modulo(zend*3600-a,3600)/60;
16 printf("  minutes %i",b);
17 c=(zend*3600-b*60-a)/3600;
18 printf("  degrees %i",c);
19    zend1=90-zend;
20     disp("altitude of star in (degrees):");
21     a=modulo(zend1*3600,60);
22 printf("seconds %.2f",a);
23 b=modulo(zend1*3600-a,3600)/60;
```

```
24 printf("  minutes %i",b);
25 c=(zend1*3600-b*60-a)/3600;
26 printf("  degrees %i",c);
27 end
```

Scilab code Exa 1.9 astronomical triangle

```
1
2
3 //exapple 1.9
4 clc; funcprot(0);
5 // Initialization of Variable
6 //solving equation of theta and delta
7 b(1,1)=90-70-20/60;//zenith distance at upper
    culmination
8 b(2,1)=90+20+40/60;//zenithdistance at lower
    culmination
9 A=[1 -1;1 1];//coeff of equations having declination
     of star and latitude of observation
10 x=A\b;
11 disp(x(1),"declination of star in (degrees)");
12 disp(x(2),"latitude of place of observation in (
    degrees):");
```

Scilab code Exa 1.10 astronomical triangle

```
1
2
3 //exapple 1.10
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14159;
7 theta=20+30/60;
```

```
H=42+6/60;//hour angle
delta=50;
//in triangle ZPM(figure in book)
PZ=(90-delta)*pi/180;
H=H*pi/180;
PM=(90-theta)*pi/180;
ZM=acos((cos(PZ)*cos(PM)+sin(PM)*sin(PZ)*cos(H)));
alpha=pi/2-ZM;
alpha=alpha*180/pi;
disp(,"altitude of star in :");
a=modulo(alpha*3600,60);
printf("seconds %.2f",a);
b=modulo(alpha*3600-a,3600)/60;
printf("  minutes %i",b);
c=(alpha*3600-b*60-a)/3600;
printf("  degrees %i",c);
A=((cos(PM)-cos(PZ)*cos(ZM))/sin(PZ)/sin(ZM));

if A<0 then
    A=-A;
    A=acos(A);
    A=180-A*180/pi;
    disp("azimuth of star in (degrees) westwards:")

   a=modulo(A*3600,60);
   printf("seconds %.2f",a);
   b=modulo(A*3600-a,3600)/60;
   printf("  minutes %i",b);
    c=(A*3600-b*60-a)/3600;
  printf("  degrees %i",c);
end
```

Scilab code Exa 1.11 astronomical triangle

```
1
```

```
2
3 //exapple 1.11
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14159;
7 theta=-8-30/60;
8 H=322;//hour angle
9 delta=50;
10 //in triangle ZPM(figure in book)
11 PZ=(90-delta)*pi/180;
12 H=2*pi-H*pi/180;
13 PM=(90-theta)*pi/180;
14 ZM=acos((cos(PZ)*cos(PM)+sin(PM)*sin(PZ)*cos(H)));
15 alpha=pi/2-ZM;
16 disp(alpha*180/pi,"altitude of star in (degrees):");
17 A=((cos(PM)-cos(PZ)*cos(ZM))/sin(PZ)/sin(ZM));
18 if A<0 then
19     A=-A;
20     A=acos(A);
21     A=180-A*180/pi;
22     disp(A,"azimuth of star in (degrees) eastwards:"
         )
23 end
```

Scilab code Exa 1.12 astronomical triangle

```
1
2
3 //exapple 1.12
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14159;
7 alpha=22+36/60;//altitude of star
8 A=42//azimuth angle
9 delta=40;//latitude of observer
```

```
10 //in triangle ZPM(figure in book)
11 PZ=(90-delta)*pi/180;
12 A=A*pi/180;
13 ZM=(90-alpha)*pi/180;
14 PM=acos((cos(PZ)*cos(ZM)+sin(ZM)*sin(PZ)*cos(A)));
15 theta=pi/2-PM;
16 disp(theta*180/pi,"declination of star in (degrees)
      north:");
17 H=((cos(ZM)-cos(PZ)*cos(PM))/sin(PZ)/sin(PM));
18 if H<0 then
19     H=-H;
20     H=acos(H);
21     H=180-H*180/pi;
22     disp(H,"hour angle  of star in (degrees)
         eastwards:")
23 end
```

Scilab code Exa 1.13 astronomical triangle

```
1
2
3 //exapple 1.13
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14159;
7 alpha=21+30/60;//altitude of star
8 A=140//azimuth angle
9 delta=48;//latitude of observer
10 //in triangle ZPM(figure in book)
11 PZ=(90-delta)*pi/180;
12 A=A*pi/180;
13 ZM=(90-alpha)*pi/180;
14 PM=(cos(PZ)*cos(ZM)+sin(ZM)*sin(PZ)*cos(A));
15 if PM<0 then
16     PM=-PM;
```

```
17      PM=acos(PM);
18      PM=180-PM*180/pi;
19      disp(PM-90,"declination of star in (degrees)
           southwards:");
20  end
21
22  H=acos((cos(ZM)-cos(PZ)*cos(PM*pi/180))/sin(PZ)/sin(
       PM*pi/180));
23  H=2*pi-H;
24  disp(H*180/pi,"hour angle of star in (degrees)");
```

Scilab code Exa 1.14 Sunset and sunrise

```
1
2
3  //exapple 1.14
4  clc; funcprot(0);
5  // Initialization of Variable
6  pi=3.14159;
7  //part1
8  delta=22+12/60;
9  theta=42+30/60;
10 ZP=(90-theta)*pi/180;
11 PM=(90-delta)*pi/180;
12 A=acos(cos(PM)/sin(ZP));
13 disp(A*180/pi,"azimuth of setting sun in (degrees):"
       );
14 H=180-acos(tan(pi/2-ZP)*tan(pi/2-PM))*180/pi;
15 disp(H,"suns hour angle in (degrees):");
16 //part2
17 delta=-22-12/60;
18 theta=42+30/60;
19 ZP=(90-theta)*pi/180;
20 PM=(180-90-delta)*pi/180;
21 A=acos(cos(PM)/sin(ZP));
```

```
22 disp(A*180/pi,"azimuth of setting sun in (degrees):"
   );
23 H=pi-acos(tan(pi/2-ZP)*tan(pi/2-PM));
24 disp(H*180/pi,"suns hour angle in (degrees):");
```

Scilab code Exa 1.15 Sunset and sunrise

```
1
2
3 //exapple 1.14
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14159;
7 //part1
8 delta=22+12/60;
9 theta=42+30/60;
10 ZP=(90-theta)*pi/180;
11 PM=(90-delta)*pi/180;
12 A=pi-acos(cos(PM)/sin(ZP));
13 disp(A*180/pi,"azimuth of setting sun in (degrees):"
   );
14 H=180-acos(tan(pi/2-ZP)*tan(pi/2-PM))*180/pi;
15 disp(H,"suns hour angle in (degrees):");
```

Scilab code Exa 1.16 Standard Time

```
1
2
3 //exapple 1.16
4 clc; funcprot(0);
5 // Initialization of Variable
6 change=-11.82/60;//change in time in a day
7 time=-3-28.41/60;//greenwich time at july1 1951
```

```
c12=change/24*12//change of time in 12 hours
tch=time-c12;
disp(tch,"greenwich mean time error in 12th hour in
   (minutes):")
```

Scilab code Exa 1.17 Standard Time

```


//exapple 1.17
clc; funcprot(0);
// Initialization of Variable
time=10*3600+30*60;
GMN=-14*60-10;
changeET=1*1.5;
neterr=GMN+changeET;
GAT=time+neterr;
hr=round(GAT/3600);
b=GAT-hr*3600;
mi=round(b/60-1);
c=GAT-hr*3600-mi*60;
disp("GAT is" );
disp(hr,"hours");
disp(mi,"minutes");
disp(c,"seconds");
```

Scilab code Exa 1.18 conversion of angle and time

```


//exapple 1.18
clc; funcprot(0);
// Initialization of Variable
```

```
6  //part1
7  A=50+12/60+48/3600;
8  time=A/15*3600
9  hr=round(time/3600);
10 b=time-hr*3600;
11 mi=round(b/60-1);
12 c=time-hr*3600-mi*60;
13 disp("time is" );
14 disp(hr,"hours");
15 disp(mi,"minutes");
16 disp(c,"seconds");
17 //part2
18 A=8+18/60+6/3600;
19 time=A/15*3600
20 hr=round(time/3600);
21 b=time-hr*3600;
22 mi=round(b/60);
23 c=time-hr*3600-mi*60;
24 disp("time is" );
25 disp(hr,"hours");
26 disp(mi,"minutes");
27 disp(c,"seconds");
28 //part3
29 A=258+36/60+30/3600;
30 time=A/15*3600
31 hr=round(time/3600);
32 b=time-hr*3600;
33 mi=round(b/60);
34 c=time-hr*3600-mi*60;
35 disp("time is" );
36 disp(hr,"hours");
37 disp(mi,"minutes");
38 disp(c,"seconds");
```

Scilab code Exa 1.19 conversion of angle and time

```
1
2
3 //exapple 1.19
4 clc; funcprot(0);
5 // Initialization of Variable
6 //part1
7 A=4+34/60+13/3600;
8 angle=A*15*3600;
9 deg=round(angle/3600-1);
10 b=angle-deg*3600;
11 mi=round(b/60);
12 c=angle-deg*3600-mi*60;
13 disp("angle is" );
14 disp(deg,"degrees");
15 disp(mi,"minutes");
16 disp(c,"seconds");
17 //part2
18 A=18+11/60+38/3600;
19 angle=A*15*3600;
20 deg=round(angle/3600-1);
21 b=angle-deg*3600;
22 mi=round(b/60-1);
23 c=angle-deg*3600-mi*60;
24 disp("angle is" );
25 disp(deg,"degrees");
26 disp(mi,"minutes");
27 disp(c,"seconds");
```

Scilab code Exa 1.20.1 local and standard time

```
1
2
3 //exapple 1.20.1
4 clc; funcprot(0);
5 // Initialization of Variable
```

```
longP=20//longitude of the place
longSM=82+30/60;//longitude of standard meridion
dolong=longSM-longP;//difference in longitude
dot=dolong/15;//difference in time
LMT=20+24/60+6/3600-dot;
disp("local mean time in past midnight:");
a=modulo(LMT*3600,60);
printf("seconds %.2f",a);
b=modulo(LMT*3600-a,3600)/60;
printf("  minutes %i",b);
c=(LMT*3600-b*60-a)/3600;
printf("  hours %i",c);
```

Scilab code Exa 1.20.2 local and standard time

```


//exapple 1.20.2
clc; funcprot(0);
// Initialization of Variable
longP=-20//longitude of the place
longSM=82+30/60;//longitude of standard meridion
dolong=longSM-longP;//difference in longitude
dot=dolong/15;//difference in time
LMT=20+24/60+6/3600-dot;
disp(LMT,"local mean time in past midnight:");
a=modulo(LMT*3600,60);
printf("seconds %.2f",a);
b=modulo(LMT*3600-a,3600)/60;
printf("  minutes %i",b);
c=(LMT*3600-b*60-a)/3600;
printf("  hours %i",c);
```

Scilab code Exa 1.21.1 Local Mean Time

```
1
2
3 //exapple 1.21.1
4 clc; funcprot(0);
5 // Initialization of Variable
6 LMT=9+40/60+12/3600;
7 longP=-42-36/60;
8 dot=longP/15;
9 GMT=LMT-dot;
10 disp("greenwich mean time in past midnight:");
11 a=modulo(GMT*3600,60);
12 printf("seconds %.2f",a);
13 b=modulo(GMT*3600-a,3600)/60;
14 printf("  minutes %i",b);
15 c=(GMT*3600-b*60-a)/3600;
16 printf("  hours %i",c);
```

Scilab code Exa 1.21.2 Local mean Time

```
1
2
3 //exapple 1.21.2
4 clc; funcprot(0);
5 // Initialization of Variable
6 LMT=4+32/60+10/3600;
7 longP=56+32/60;
8 dot=longP/15;
9 GMT=LMT-dot;
10 disp("greenwich mean time in past midnight:");
11 a=modulo(GMT*3600,60);
12 printf("seconds %.2f",a);
13 b=modulo(GMT*3600-a,3600)/60;
14 printf("  minutes %i",b);
```

```
15 c=(GMT*3600-b*60-a)/3600;
16 printf("  hours %i",c);
```

Scilab code Exa 1.22.1 GCT to LMT

```
1
2
3 //exapple 1.22.1
4 clc; funcprot(0);
5 // Initialization of Variable
6 GCT=18+40/60+12/3600;//greenwich civil time
7 longP=72+30/60;//longitude of the place
8 dot=longP/15;
9 LMT=GCT+dot;
10 disp("local mean time in past midnight:");
11 a=modulo(LMT*3600,60);
12 printf("seconds %.2f",a);
13 b=modulo(LMT*3600-a,3600)/60;
14 printf("  minutes %i",b);
15 c=(LMT*3600-b*60-a)/3600;
16 printf("  hours %i",c);
```

Scilab code Exa 1.22.2 GCT to LMT

```
1
2
3 //exapple 1.22.2
4 clc; funcprot(0);
5 // Initialization of Variable
6 GCT=18+40/60+12/3600;//greenwich civil time
7 longP=-73+30/60;//longitude of the place
8 dot=longP/15;
9 LMT=GCT+dot;
```

```
10 disp("local mean time in past midnight:");
11 a=modulo(LMT*3600,60);
12 printf("seconds %.2f",a);
13 b=modulo(LMT*3600-a,3600)/60;
14 printf("  minutes %i",b);
15 c=(LMT*3600-b*60-a)/3600;
16 printf("  hours %i",c);
```

Scilab code Exa 1.22.3 GCT to LMT

```
1
2
3 //exapple 1.22.3
4 clc; funcprot(0);
5 // Initialization of Variable
6 GCT=18+40/60+12/3600;//greenwich civil time
7 longP=102+30/60;//longitude of the place
8 dot=longP/15;
9 LMT=GCT+dot;
10 disp("local mean time in past midnight:");
11 a=modulo(LMT*3600,60);
12 printf("seconds %.2f",a);
13 b=modulo(LMT*3600-a,3600)/60;
14 printf("  minutes %i",b);
15 c=(LMT*3600-b*60-a)/3600;
16 if c>24 then
17     c=c-24;
18 end
19 printf("  hours %i",c);
20 disp("date is changed by a day");
21 disp("date at the place is 3th July")
```

Scilab code Exa 1.23 Actual Time

```
1
2
3 //exapple 1.23
4 clc; funcprot(0);
5 // Initialization of Variable
6 LMT=10+20/60+30/3600;//local mean time
7 longP=102+30/60;//longitude of the place
8 dot=longP/15;
9 GMT=LMT-dot;
10 mGMN=12-GMT;//mean time interval
11 i=mGMN*0.32/3600;//increase in mGMN
12 ETGMN=5/60+4.35/3600;
13 ch=i+ETGMN;//change in GMT
14 GMT=ch+GMT;
15 LMT=GMT+dot;
16 disp("local mean time in past midnight observed:");
17 a=modulo(LMT*3600,60);
18 printf("seconds %.2f",a);
19 b=modulo(LMT*3600-a,3600)/60;
20 printf("  minutes %i",b);
21 c=(LMT*3600-b*60-a)/3600;
22 if c>24 then
23     c=c-24;
24 end
25 printf("  hours %i",c);
```

Scilab code Exa 1.24 LMT

```
1
2
3 //exapple 1.24
4 clc; funcprot(0);
5 // Initialization of Variable
6 LMT=15+12/60+40/3600;//local mean time
7 longP=-20-30/60;//longitude of the place
```

```
dot=longP/15;
GMT=LMT-dot;
mGMN=GMT-12;//mean time interval
i=mGMN*0.22/3600;//increase in mGMN
ETGMN=5/60+10.65/3600;
ch=i+ETGMN;//change in GMT
GMT=ch+GMT;
LMT=GMT+dot;
disp("local mean time in past midnight observed:");
a=modulo(LMT*3600,60);
printf("seconds %.2f",a);
b=modulo(LMT*3600-a,3600)/60;
printf("  minutes %i",b);
c=(LMT*3600-b*60-a)/3600;
if c>24 then
    c=c-24;
end
printf("  hours %i",c);
```

Scilab code Exa 1.25 Local time

```


//exapple 1.25
clc; funcprot(0);
// Initialization of Variable
time=4+20/60+30/3600;
accn=time*9.8565/3600;//acceleration
stime=time+accn;//sideral time
disp("local mean time in past midnight observed:");
a=modulo(stime*3600,60);
printf("seconds %.3f",a);
b=modulo(stime*3600-a,3600)/60;
printf("  minutes %i",b);
c=(stime*3600-b*60-a)/3600;
```

```
15 if c>24 then
16     c=c-24;
17 end
18 printf("  hours %i",c);
```

Scilab code Exa 1.26 Local time

```
1
2
3 //exapple 1.26
4 clc; funcprot(0);
5 // Initialization of Variable
6 time=8+40/60+50/3600;
7 accn=-time*9.8565/3600;//acceleration
8 stime=time+accn;//sideral time
9 disp("local mean time in past midnight observed:");
10 a=modulo(stime*3600,60);
11 printf("seconds %.3f",a);
12 b=modulo(stime*3600-a,3600)/60;
13 printf("  minutes %i",b);
14 c=(stime*3600-b*60-a)/3600;
15 if c>24 then
16     c=c-24;
17 end
18 printf("  hours %i",c);
```

Scilab code Exa 1.27.1 GST of GMN

```
1
2
3 //exapple 1.27.1
4 clc; funcprot(0);
5 // Initialization of Variable
```

```
longP=-160-30/60-30/3600;//longitude of place
GST=16+30/60+12/3600;//standard time
dot=longP/15;//difference in time
i=dot*9.8565/3600;//error
LST=GST-i;
disp("local mean time in past midnight observed:");
a=modulo(LST*3600,60);
printf("seconds %.2f",a);
b=modulo(LST*3600-a,3600)/60;
printf("  minutes %i",b);
c=(LST*3600-b*60-a)/3600;
if c>24 then
    c=c-24;
end
printf("  hours %i",c);
```

Scilab code Exa 1.27.2 GST at GMM

```


//exapple 1.27.2
clc; funcprot(0);
// Initialization of Variable
longP=160+30/60+30/3600;//longitude of place
GST=16+30/60+12/3600;//standard time
dot=longP/15;//difference in time
i=dot*9.8565/3600;//error
LST=GST-i;
disp("local mean time in past midnight observed:");
a=modulo(LST*3600,60);
printf("seconds %.2f",a);
b=modulo(LST*3600-a,3600)/60;
printf("  minutes %i",b);
c=(LST*3600-b*60-a)/3600;
if c>24 then
```

```
18      c=c-24;
19  end
20  printf("  hours %i",c);
```

Scilab code Exa 1.28 LST at LMT

```
1
2
3  //exapple 1.28
4  clc; funcprot(0);
5  // Initialization of Variable
6  longP=85+20/60;//longitude of place
7  GST=6+30/60;//standard time
8  GMN=6+32/60+12/3600;
9  dot=longP/15;//difference in time
10 i=dot*9.8565/3600;//error
11 LST=GMN-i;//LST at L.M.N
12 i2=GST*9.8565/3600;//error in GST
13 LST2=GST+i2;
14 LST=LST+LST2//lst at L.M.N
15 disp("local mean time in past midnight observed:");
16 a=modulo(LST*3600,60);
17 printf("seconds %.2f",a);
18 b=modulo(LST*3600-a,3600)/60;
19 printf("  minutes %i",b);
20 c=(LST*3600-b*60-a)/3600;
21 if c>24 then
22      c=c-24;
23 end
24 printf("  hours %i",c);
```

Scilab code Exa 1.29 LST and LMT

```
1
2
3 //exapple 1.29
4 clc; funcprot(0);
5 // Initialization of Variable
6 longP=112+20/60+15/3600;//longitude of place
7 GST=8+10/60+28/3600;//GST at GMM
8 lst=18+28/60+12/3600;//local sidereal time
9 dot=longP/15;//difference in time
10 i=dot*9.8565/3600;//error
11 LST=GST+i;//LST at L.M.N
12 LMM=lst-LST;
13 i2=LMM*9.8565/3600;//error in LMM
14 LMT=LMM-i2;//local mean time
15
16 disp("local mean time in past midnight observed:");
17 a=modulo(LMT*3600,60);
18 printf("seconds %.2f",a);
19 b=modulo(LMT*3600-a,3600)/60;
20 printf("  minutes %i",b);
21 c=(LMT*3600-b*60-a)/3600;
22 if c>24 then
23     c=c-24;
24 end
25 printf("  hours %i",c);
```

Scilab code Exa 1.30 LST at LMT

```
1
2
3 //exapple 1.30
4 clc; funcprot(0);
5 // Initialization of Variable
6 longP=85+20/60;//longitude of place
7 GST=18+30/60;//standard time
```

```
gst=6+32/60+12/3600;//GST at GMN
dot=longP/15;//difference in time
GMT=GST-dot-12;
i=GMT*9.8565/3600;//error
GMT=GMT+i;//SI time
LST=GMT+dot+gst;//LST at LMT
disp("local standard time in past midnight observed:
   ");
a=modulo(LST*3600,60);
printf("seconds %.2f",a);
b=modulo(LST*3600-a,3600)/60;
printf("  minutes %i",b);
c=(LST*3600-b*60-a)/3600;
if c>24 then
    c=c-24;
end
printf("  hours %i",c);
```

Scilab code Exa 1.31 LST and LMT

```

//exapple 1.31
clc; funcprot(0);
// Initialization of Variable
longP=112+20/60+15/3600;//longitude of place
GST=8+10/60+28/3600;//GST at GMM
lst=18+28/60+12/3600;//local sidereal time
dot=longP/15;//difference in time
gmm=lst+dot-GST;//SI at GMM

i=gmm*9.8565/3600;//error
gmm=gmm-i;//LST at L.M.N
LMT=gmm-dot;//local mean time
disp("local mean time in past midnight observed:");
```

```
a=modulo(LMT*3600,60);
printf("seconds %.2f",a);
b=modulo(LMT*3600-a,3600)/60;
printf("  minutes %i",b);
c=(LMT*3600-b*60-a)/3600;
if c>24 then
    c=c-24;
end
printf("  hours %i",c);
```

Scilab code Exa 1.32 LST and LMT

```
1
2
3
4
5 //exapple 1.32
6 clc; funcprot(0);
7 // Initialization of Variable
8 //part1
9 longP=162+30/60+15/3600;//longitude of place
10 GST=10+30/60+15/3600;//GST at GMN
11 RA=22+11/60+30/3600;//local sidereal time
12 dot=longP/15;//difference in time
13 i=dot*9.8565/3600;//error
14 gmm=GST+i;//LST at L.M.N
15 lmn=RA-gmm;//SI of LMN
16 i2=lmn*9.8565/3600;//error 2
17 LMT=lmn-i2;
18 disp("local mean time in past midnight observed at
   upper transit:");
19 a=modulo(LMT*3600,60);
20 printf("seconds %.2f",a);
21 b=modulo(LMT*3600-a,3600)/60;
22 printf("  minutes %i",b);
```

```
23 c=(LMT*3600-b*60-a)/3600;
24 if c>24 then
25     c=c-24;
26 end
27 printf("  hours %i",c);
28 //part2
29 i3=12*9.8565/3600;//ratardation
30 LMT=LMT+12-i3;
31 disp("local mean time in past midnight observed at
      lower transit:");
32 a=modulo(LMT*3600,60);
33 printf("seconds %.2f",a);
34 b=modulo(LMT*3600-a,3600)/60;
35 printf("  minutes %i",b);
36 c=(LMT*3600-b*60-a)/3600;
37 if c>24 then
38     c=c-24;
39 end
40 printf("  hours %i",c);
41 disp("its the time of next day")
```

Scilab code Exa 1.33 LST and LMT

```
1
2
3
4
5 //exapple 1.33
6 clc; funcprot(0);
7 // Initialization of Variable
8 longP=60+30/60;//longitude of place
9 GST=7+30/60+48.6/3600;//GST at GMM
10 RA=17+28/60/40/1600;
11 dot=longP/15;//difference in time
12 i=dot*9.8565/3600;//error
```

```
13 gmm=GST-i;//LST at L.M.N
14 LMT=RA-gmm;//local mean time
15 disp("local mean time in past midnight observed:");
16 a=modulo(LMT*3600,60);
17 printf("seconds %.2f",a);
18 b=modulo(LMT*3600-a,3600)/60;
19 printf("  minutes %i",b);
20 c=(LMT*3600-b*60-a)/3600;
21 if c>24 then
22     c=c-24;
23 end
24 printf("  hours %i",c);
```

Scilab code Exa 1.42 horizontal angle

```
1
2
3 //exapple 1.42
4 clc; funcprot(0);
5 // Initialization of Variable
6 aziA=32+41/60+30/3600;//azimuth of A
7 aziB=110+28/60+42/3600;//azimuth of B
8 vaA=10+21/60+12/3600;//vertical angle of A
9 vaB=-2-18/60-30/3600;//verticsl angle of B
10 lA1=11;
11 lB1=11.5;
12 rA1=7.5;
13 rB1=7;
14 lB2=10;
15 lA2=10.5;
16 rB2=7.5;
17 rA2=8;
18 d=20;
19 //partA
20 sigl=lA1+lA2;
```

```
21 sigr=rA1+rA2;
22 b=sigl/4*d-sigr/4*d;
23 i=tan(vaA);
24 caziA=aziA+i*29.95/3600;
25 disp(caziA,"corrected azimuth of A in (degrees):");
26 //partB
27
28 i=tan(vaB);
29 caziB=aziB+i*b/3600;
30 disp(caziB,"corrected azimuth of B in (degrees):");
31 ha=caziB-caziA;
32 disp(ha,"horizontal difference of angle between A &
     B in (degrees):")
```

Scilab code Exa 1.43 index error in theodolite

```
1
2
3 //exapple 1.43
4 clc; funcprot(0);
5 // Initialization of Variable
6 v1=18+36/60+48/3600;//vertical angle 1
7 v2=18+35/60+56/3600;//vertical angle 2
8 slm=28+36/60+20/3600;//altitude of sun measured
9 ds=15/60+59.35/3600;//dia of sun
10 pi=3.14159;
11 mv=(v1+v2)/2;//mean vertical angle
12 i=v1-v2;//error
13 sl=slm+i;//new altitude of sun
14 //part2
15 sl=sl+ds;
16 //part3
17 ir=-57/3600/(tan(slm*pi/180+26*pi/180/3600));//error
       due to refraction
18 sl=sl+ir;
```

```
19 //part4
20 ip=8.8/3600*cos(slm*pi/180+26*pi/180/3600);//error
     due to parallax
21 sl=sl+ip;
22 disp("correct altitude of sun")
23 a=modulo(sl*3600,60);
24 printf("seconds %.2f",a);
25 b=modulo(sl*3600-a,3600)/60;
26 printf("  minutes %i",b);
27 c=(sl*3600-b*60-a)/3600;
28 printf("  degrees %i",c);
```

Scilab code Exa 1.44 chronometer error

```
1
2
3 //exapple 1.44
4 clc; funcprot(0);
5 // Initialization of Variable
6 long=4+30/60;
7 i=long*9.8565/3600;//longitude
8 gst=14+38/60+12/3600;//GST on GMM
9 lst=gst-i;//LST on LMM
10 RA=7+36/60+21.24/3600;
11 LST=RA;
12 SI=LST-lst+24;
13 LCT=17+56/60+8.86/3600-1;//local chronometer time
14 i2=SI*9.8296/3600;
15 LMM=SI-i2;
16 ce=LCT-LMM;
17 disp(ce*3600,"chronometer error in (s):")
```

Scilab code Exa 1.45 chronometer error

```
1
2
3 //exapple 1.45
4 clc; funcprot(0);
5 // Initialization of Variable
6 c=90-36-30/60-30/3600;//co latitude
7 p=90-16-12/60-18.4/3600;//co declination
8 z=90-30-12/60-30/3600;//co altitude
9 s=(p+z+c)/2;
10 pi=3.14159;
11 s1=s-c;
12 s2=s-p;
13 s3=s-z;
14 H=2*atan(sqrt(sin(s1*pi/180)*sin(s2*pi/180)/sin(s*pi
     /180)/sin(s3*pi/180)));
15 H=H*180/pi;
16 H=24-H/15;
17 LST=H+5+18/60+12.45/3600-24;
18 ce=1+2/60+5.25/3600-LST;
19 disp(ce*3600+2,"chronometer error in (s):")
```

Scilab code Exa 1.46 chronometer error

```
1
2
3 //exapple 1.46
4 clc; funcprot(0);
5 // Initialization of Variable
6 c=90-36-40/60-30/3600;//co latitude
7 p=90-17-26/60-42.1/3600;//co declination
8 z=90-36-14/60-16.8/3600;//co altitude
9 pi=3.14159;
10 s=(p+z+c)/2;
11 s1=s-c;
12 s2=s-p;
```

```
13 s3=s-z;
14 H=2*atan(sqrt(sin(s1*pi/180)*sin(s2*pi/180)/sin(s*pi
      /180)/sin(s3*pi/180)));
15 H=H*180/pi;
16 H=H/15;
17 i=12-11-56/60-22.8/3600;//error in time
18 LAT=15+49/60+40.6/3600;//local actual time
19 GAT=LAT-H;
20 GMT=GAT-i;
21 LMT=GMT+H;
22 ce=15+49/60+12.6/3600-LMT;
23 disp(ce*3600,"chronometer error in (s):")
```

Scilab code Exa 1.47 chronometer error

```
1
2
3 //exapple 1.47
4 clc; funcprot(0);
5 // Initialization of Variable
6 RA=17+12/60+48/3600;
7 gst=9+26/60+12/3600;//GST on GMN
8 long=138/15+45/15/60;//longitude
9 lst=-long*9.85645/3600+9+26/60+12/3600;//LST on LMN
10 LST=17+12/60+48/3600;//local sidereal time
11 SI=LST-lst;
12 MI=-SI*9.8296/3600+SI;
13 LCT=7+47/60+02/3600;//local chronometer time
14 ce=LCT-MI;
15 disp(ce*3600,"chronometer error in (s):")
```

Scilab code Exa 1.48 Azimuth and LMT

```
1
2
3  //exapple 1.48
4  clc; funcprot(0);
5  // Initialization of Variable
6  //part1
7  theta=54+30/60;//logitude
8  delta=62+12/60+21/3600;//declination
9  pi=3.14;
10 lat=asin(sin(theta*pi/180)/sin(delta*pi/180));
11 lat=lat*180/pi;
12 disp("latitude of star observed:");
13 a=modulo(lat*3600,60);
14 printf("seconds %.2f",a);
15 b=modulo(lat*3600-a,3600)/60;
16 printf("  minutes %i",b);
17 c=(lat*3600-b*60-a)/3600;
18 printf("  degrees %i",c);
19 //part2
20 A=53+25/60;//azimuth of star
21 h=65+18/60+42/3600;//horizontal angle
22 A=A+h;
23 disp("azimuth of line AP is:");
24 a=modulo(A*3600,60);
25 printf("seconds %.2f",a);
26 b=modulo(A*3600-a,3600)/60;
27 printf("  minutes %i",b);
28 c=(A*3600-b*60-a)/3600;
29 printf("  degrees %i",c);
30 //part3
31 lst=4+39/60+6.5/3600;//LST of LMN
32 LST=10+58/60+38/3600+2+49/60+25.3/3600;//LST of
      observation
33 LMN=LST-lst;
34 i=LMN*9.8565/3600;//error
35 LMT=LMN-i;
36 disp("local mean time is:")
37 a=modulo(LMT*3600,60);
```

```
38 printf("seconds %.2f",a);
39 b=modulo(LMT*3600-a,3600)/60;
40 printf("  minutes %i",b);
41 c=(LMT*3600-b*60-a)/3600;
42 printf("  hours %i",c);
```

Scilab code Exa 1.49 Azimuth and LMT

```
1
2
3 //exapple 1.49
4 clc; funcprot(0);
5 // Initialization of Variable
6 //part1
7 theta=53+32/60;//logitude
8 delta=56+42/60+53.2/3600;//declination
9 pi=3.14159;
10 lat=asin(sin(theta*pi/180)/sin(delta*pi/180));
11 lat=lat*180/pi;
12 disp("altitude of star observed:");
13 a=modulo(lat*3600,60);
14 printf("seconds %.2f",a);
15 b=modulo(lat*3600-a,3600)/60;
16 printf("  minutes %i",b);
17 c=(lat*3600-b*60-a)/3600;
18 printf("  degrees %i",c);
19 //part2
20 As=asin(cos(delta*pi/180)/cos(theta*pi/180));//
     azimuth of star
21 h=75+18/60+20/3600;//angle between line and star
22 A=h-As*180/pi;
23 A=360-A;
24 disp("azimuth of line AP is:");
25 a=modulo(A*3600,60);
26 printf("seconds %.2f",a);
```

```
27 b=modulo(A*3600-a,3600)/60;
28 printf("  minutes %i",b);
29 c=(A*3600-b*60-a)/3600;
30 printf("  degrees %i",c);
31 //part3
32 LST=10+58/60+3.9/3600+22+10/60+38.5/3600-24;//LST of
       observation
33 long=5+40/60+18/3600;//longitude
34 i=long*9.8565/3600;//error
35 lst=4+58/60+23.84/3600+i;//LST on LMN
36 LMM=LST-lst;
37 i2=LMM*9.8565/3600;//error in LMM
38 LMT=LMM-i2;
39 disp("local mean time interval is:")
40 a=modulo(LMT*3600,60);
41 printf("seconds %.2f",a);
42 b=modulo(LMT*3600-a,3600)/60;
43 printf("  minutes %i",b);
44 c=(LMT*3600-b*60-a)/3600;
45 printf("  hours %i",c);
```

Scilab code Exa 1.50 Azimuth of star

```
1
2
3 //exapple 1.50
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14159;
7 long=(15+30/60)/15;//longitude
8 GMT=19+12/60+36.2/3600;
9 i=long*9.8565/3600;//error in longitude
10 LMT=GMT+long;
11 i2=LMT*9.8565/3600;//error in LMT
12 LMT=LMT+i2;
```

```
13 LST=10+12/60+21.4/3600+LMT;
14 H=LST-10-12/60-6.3/3600;//hour angle
15 H=H*15;
16 H=360-H;
17 As=atan(tan((55+29/60+1.2/3600)*pi/180)*cos
      ((32+52/60+27/3600)*pi/180)/sin((-2-7/60-33/3600)
      *pi/180));
18 h=36+28/60+18/3600;//angle between line and star
19 A=180+As*180/pi-h;
20 disp("azimuth of star is:");
21 a=modulo(A*3600,60);
22 printf("seconds %.2f",a);
23 b=modulo(A*3600-a,3600)/60;
24 printf("  minutes %i",b);
25 c=(A*3600-b*60-a)/3600;
26 printf("  degrees %i",c);
27 clear
```

Scilab code Exa 1.51 Azimuth of line

```
1
2
3 //exapple 1.51
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14159;
7 alpha=33+35/60+10/3600;//altitude
8 ZM=90-alpha;
9 delta=22+05/60+35/3600;//declination
10 PM=90-delta;
11 theta=52+30/60+20/3600;//latitude
12 ZP=90-theta;
13 As=acos((cos(PM*pi/180)-cos(ZP*pi/180)*cos(ZM*pi
      /180))/(sin(ZP*pi/180)*sin(ZM*pi/180)));
14 h=18+20/60+30/3600;//angle between line and star
```

```
A=As*180/pi+h;
disp("azimuth of star is:");
a=modulo(A*3600,60);
printf("seconds %.2f",a);
b=modulo(A*3600-a,3600)/60;
printf("  minutes %i",b);
c=(A*3600-b*60-a)/3600;
printf("  degrees %i",c);
clear
```

Scilab code Exa 1.52 Azimuth of sun

```
1
2
//exapple 1.52
clc; funcprot(0);
// Initialization of Variable
//part1
GAT=5+17/60+6/60;//GAT of observation
delta=17+46/60+52/3600;//declination
i=37/3600*GAT;
delta=delta-i;
disp("declination of GAT is:");
a=modulo(delta*3600,60);
printf("seconds %.2f",a);
b=modulo(delta*3600-a,3600)/60;
printf("  minutes %i",b);
c=(delta*3600-b*60-a)/3600;
printf("  degrees %i",c);
//part2
pi=3.14159;
p=90-delta;//co-declination
altitude=23+15/60+20/3600;//altitude of sun
i2=2/60+12/3600;//error due to refraction
i3=8/3600;//error due to parallax
```

```
24 altitude=altitude-i2+i3;
25 c=90-55-46/60-12/3600;//colatitude
26 z=90-altitude;//co altitude
27 s=(p+z+c)/2;
28 s1=s-c;
29 s2=s-p;
30 s3=s-z;
31 A=2*atan(sqrt(sin(s3*pi/180)*sin(s1*pi/180)/sin(s*pi
      /180)/sin(s2*pi/180)));
32 A=A*180/pi;
33 disp("azimuth of star is:");
34 a=modulo(A*3600,60);
35 printf("seconds %.2f",a);
36 b=modulo(A*3600-a,3600)/60;
37 printf("  minutes %i",b);
38 c=(A*3600-b*60-a)/3600;
39 printf("  degrees %i",c);
40 clear
```

Scilab code Exa 1.53 azimuth angle

```
1
2
3 //exapple 1.53
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14159;
7 GMT=17+5/60+2/3600;//
8 i=9.8565/3600*GMT;
9 GST=3+12/60+12/3600;
10 wl=1+18/60;//west longitude
11 RA=16+23/60+30/3600;
12 H=GMT+i+GST+wl-RA;//hour angle
13 H=H*15;
14 p=90-29-52/60;
```

```
15 c=90-52-8/60;
16 z=acos(cos(H*pi/180)*sin(p*pi/180)*sin(c*pi/180)+cos
      (p*pi/180)*cos(c*pi/180));
17 A=asin(sin(p*pi/180)*sin(H*pi/180)/sin(z));
18 A=A*180/pi;
19 disp("azimuth of star is:");
20 a=modulo(A*3600,60);
21 printf("seconds %.2f",a);
22 b=modulo(A*3600-a,3600)/60;
23 printf("  minutes %i",b);
24 c=(A*3600-b*60-a)/3600;
25 printf("  degrees %i",c);
26 clear
```

Scilab code Exa 1.54 azimuth angle

```
1
2
3 //exapple 1.54
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14159;
7 a1=24+30/60+20/3600;
8 a2=24+30/60+40/3600;
9 a3=25;
10 a4=25+1/60;
11 lat=(a1+a2+a3+a4)/4;
12 il=(10.6-9.4)/4*15/3600;//error due to level
13 lat=lat+il;
14 ir=-57/3600/tan((lat*pi/180));//carrection of
      refraction
15 ip=8/3600*cos(lat*pi/180);//correction of parallax
16 lat=lat-ir+ip;
17 z=90-lat;
18 delta=1+32/60+16.8/3600-56.2/3600*(3/60+1.86/3600);
```

```
19 p=90-delta;
20 c=90-36-48/60-30/3600;
21 s=(p+z+c)/2;
22 s1=s-c;
23 s2=s-p;
24 s3=s-z;
25 A=2*atan(sqrt(sin(s3*pi/180)*sin(s1*pi/180)/sin(s*pi
      /180)/sin(s2*pi/180)));
26 A=A*180/pi;
27
28 A=A+81+59/60+10/3600;
29 A=360-A;
30 disp("azimuth of star is:");
31 a=modulo(A*3600,60);
32 printf("seconds %.2f",a);
33 b=modulo(A*3600-a,3600)/60;
34 printf("  minutes %i",b);
35 c=(A*3600-b*60-a)/3600;
36 printf("  degrees %i",c);
37 clear
```

Scilab code Exa 1.55 latitude

```
1
2
3 //exapple 1.55
4 clc; funcprot(0);
5 // Initialization of Variable
6 alpha=65+40/60+18/3600;//altitude
7 delta=53+12/60+10/3600;//declination
8 pi=3.1415;
9 i=57/3600*1/tan(alpha*pi/180);//error
10 alpha=alpha-i;
11 z=90-alpha;//zenith distance
12 lat=delta-z;
```

```
13 disp("latitude of star observed:");
14 a=modulo(lat*3600,60);
15 printf("seconds %.2f",a);
16 b=modulo(lat*3600-a,3600)/60;
17 printf("  minutes %i",b);
18 c=(lat*3600-b*60-a)/3600;
19 printf("  degrees %i",c);
20 clear
```

Scilab code Exa 1.56 latitude

```
1
2
3 //exapple 1.56
4 clc; funcprot(0);
5 // Initialization of Variable
6 alpha=64+36/60+20/3600;//altitude
7 delta=26+12/60+10/3600;//declination
8 pi=3.1415;
9 i=57/3600*1/tan(alpha*pi/180);//error
10 alpha=alpha-i;
11 z=90-alpha;//zenith distance
12 lat=delta+z;
13 disp("latitude of star observed:");
14 a=modulo(lat*3600,60);
15 printf("seconds %.2f",a);
16 b=modulo(lat*3600-a,3600)/60;
17 printf("  minutes %i",b);
18 c=(lat*3600-b*60-a)/3600;
19 printf("  degrees %i",c);
20 clear
```

Scilab code Exa 1.57 latitude of place

```
1
2
3 //exapple 1.57
4 clc; funcprot(0);
5 // Initialization of Variable
6 alpha=44+12/60+30/3600;//altitude
7 longP=75+20/60+15/3600;//longitude of place
8 delta=22+18/60+12.8/3600;//declination of sun
9 pi=3.1415;
10 i=57/3600*1/tan(alpha*pi/180);//error
11 i2=8.78/3600*cos(alpha);//correction due to parallax
12 i3=15/60+45.86/3600;//coreectin due to semi diamter
13 alpha=alpha-i+i2+i3;
14 z=90-alpha;//zenith distance
15 delT=longP/15;
16 i4=6.82/3600*delT;//error in time
17 delta=i4+delta;
18 lat=delta+z;
19 disp("latitude of sun is to the south observed:");
20 a=modulo(lat*3600,60);
21 printf("seconds %.2f",a);
22 b=modulo(lat*3600-a,3600)/60;
23 printf("  minutes %i",b);
24 c=(lat*3600-b*60-a)/3600;
25 printf("  degrees %i",c);
26 clear
```

Scilab code Exa 1.58 apparent latitude of place

```
1
2
3 //exapple 1.58
4 clc; funcprot(0);
5 // Initialization of Variable
6 theta=80;
```

```
delta=46+45/60+30/3600;
alpha1=90-theta+delta;
disp("alpha1 is (North):");
a=modulo(alpha1*3600,60);
printf("seconds %.2f",a);
b=modulo(alpha1*3600-a,3600)/60;
printf("  minutes %i",b);
c=(alpha1*3600-b*60-a)/3600;
printf("  degrees %i",c);
alpha2=theta+delta-90;
disp("alpha2 is (South):");
a=modulo(alpha2*3600,60);
printf("seconds %.2f",a);
b=modulo(alpha2*3600-a,3600)/60;
printf("  minutes %i",b);
c=(alpha2*3600-b*60-a)/3600;
printf("  degrees %i",c);
```

Scilab code Exa 1.59 latitude of place

```

//exapple 1.59
clc; funcprot(0);
// Initialization of Variable
delta1=20+25/60+48/3600;//declination of star 1
delta2=79+30/60+52/3600;//declination of star 2
alpha1=48+18/60+12/3600;//altitude of star 1
alpha2=47+54/60+6/3600;//altitude of star 2
pi=3.1415;
r1=58/3600/tan(alpha1*pi/180)//error 1
r2=58/3600/tan(alpha2*pi/180)//error 2
lat=90-(alpha1-alpha2)/2+(delta1-delta2)/2+(r1-r2)
   /2;
disp("latitude of star observed:");
```

```
15 a=modulo(lat*3600,60);
16 printf("seconds %.2f",a);
17 b=modulo(lat*3600-a,3600)/60;
18 printf("  minutes %i",b);
19 c=(lat*3600-b*60-a)/3600;
20 printf("  degrees %i",c);
```

Scilab code Exa 1.60 latitude of place and declination of star

```
1
2
3 //exapple 1.60
4 clc; funcprot(0);
5 // Initialization of Variable
6 alphal=18+36/60+40/3600;//altitude at lower
     culmination
7 alphau=59+48/60+20/3600;//altitude at upper
     culmination
8 lat=(alphal+alphau)/2;
9 disp("latitude of star observed:");
10 a=modulo(lat*3600,60);
11 printf("seconds %.2f",a);
12 b=modulo(lat*3600-a,3600)/60;
13 printf("  minutes %i",b);
14 c=(lat*3600-b*60-a)/3600;
15 printf("  degrees %i",c);
16 delta=90+lat-alphau;
17 disp("declination of star observed:");
18 a=modulo(delta*3600,60);
19 printf("seconds %.2f",a);
20 b=modulo(delta*3600-a,3600)/60;
21 printf("  minutes %i",b);
22 c=(delta*3600-b*60-a)/3600;
23 printf("  degrees %i",c);
```

Scilab code Exa 1.61 latitude of place

```
1
2
3 //exapple 1.61
4 clc; funcprot(0);
5 // Initialization of Variable
6 alpha=40+36/60+30/3600;//altitude of star
7 delta=10+36/60+40/3600;//declination of star
8 H=46+36/60+20/3600;//hour angle of star
9 pi=3.1412;
10 n=atan(tan(delta*pi/180)/cos(H*pi/180));
11 lat=n+acos(sin(alpha*pi/180)*sin(n)/sin(delta*pi
     /180));
12 lat=lat*180/pi;
13 disp("latitude of star observed:");
14 a=modulo(lat*3600,60);
15 printf("seconds %.2f",a);
16 b=modulo(lat*3600-a,3600)/60;
17 printf("  minutes %i",b);
18 c=(lat*3600-b*60-a)/3600;
19 printf("  degrees %i",c);
```

Scilab code Exa 1.62 latitude of place

```
1
2
3 //exapple 1.62
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14159;
7 alpha=42+10/60+40/3600;//altitude of sun
```

```
8  delta=23+12/60+18.6/3600;//declination of sun's
       angle
9  i=57/3600*1/tan(alpha*pi/180);//error
10 i2=8.78/3600*cos(alpha);//correction due to parallax
11 i3=15/60+45.86/3600;//coreectin due to semi diamter
12 longP=108+30/60;//longitude of place
13 LMT=14+50/60;
14 alpha=alpha-i+i2+i3;
15 delT=longP/15;//change in time
16 GMT=LMT+delT;
17 i4=1.2/3600*GMT;//error in time
18 H=(GMT-12+i4-delT)*15;//hour angle
19 i5=10.6/3600*GMT;//error in declination
20 delta=delta+i5;
21 ZM=(90-alpha)*pi/180;
22 PM=(90+delta)*pi/180;
23 A=asin(sin(PM)/sin(ZM)*sin(H*pi/180));
24 A=pi-A;
25 ZP=2*atan(sin(A/2+H*pi/360)/sin(A/2-H*pi/360)*tan(PM
       /2-ZM/2));
26 lat=pi/2-ZP;
27 lat=lat*180/pi+1;
28 disp("latitude of star observed:");
29 a=modulo(lat*3600,60);
30 printf("seconds %.2f",a);
31 b=modulo(lat*3600-a,3600)/60;
32 printf("  minutes %i",b);
33 c=(lat*3600-b*60-a)/3600;
34 printf("  degrees %i",c);
```

Scilab code Exa 1.63 latitude of place

```
1
2
3 //exapple 1.63
```

```
4  clc; funcprot(0);
5  // Initialization of Variable
6  delta=15+20/60+48/3600;//declination of star
7  Int=9+22/60+6/3600;//interval
8  pi=3.141;
9  dint=Int*9.8565/3600;//change in interval
10 H=(Int+dint)*15/2;//hour angle
11 lat=atan(tan(delta*pi/180)/cos(H*pi/180));
12 lat=lat*180/pi+5/6*16/3600;
13 disp("latitude of star observed:");
14 a=modulo(lat*3600,60);
15 printf("seconds %.2f",a);
16 b=modulo(lat*3600-a,3600)/60;
17 printf("  minutes %i",b);
18 c=(lat*3600-b*60-a)/3600;
19 printf("  degrees %i",c);
```

Scilab code Exa 1.64 latitude of place

```
1
2
3  //exapple 1.64
4  clc; funcprot(0);
5  // Initialization of Variable
6  pi=3.14159;
7  RA=1+41/60+48.64/3600;
8  lat=48+36/60+40/3600;//latitude
9  delta=88+58/60+28.26/3600;//declination of polaris
10 GMM=16+48/60+20.86/3600;
11 longP=7+20/60;//longitude of place P
12 i1=51/3600;//error due to barometer
13 i2=1/3600;//error due to barometer
14 i3=-1/3600;//error due to temp
15 lat=lat-i1+i2+i3;
16 delT=longP/15;
```

```
17 i4=delT*9.8565/3600;
18 lst=GMM+i4;
19 LMT=20+24/60+50/3600;
20 i6=9.8565/3600*LMT;//error in LMT
21 LST=LMT+i6+lst-24;
22 H=LST-RA;//hour angle
23 H=H*15;
24 lat=lat-(90-delta)*cos(H*pi/180)+.5*sin(1/3600*pi
      /180)*(90-delta)^2*(sin(H*pi/180))^2*tan(lat*pi
      /180);
25 disp("latitude of star observed:");
26 a=modulo(lat*3600,60);
27 printf("seconds %.2f",a);
28 b=modulo(lat*3600-a,3600)/60;
29 printf("  minutes %i",b);
30 c=(lat*3600-b*60-a)/3600;
31 printf("  degrees %i",c);
```

Scilab code Exa 1.65 latitude of place

```
1
2
3 //exapple 1.65
4 clc; funcprot(0);
5 // Initialization of Variable
6 longP=120-4-20/60;//longitude of point
7 delT=longP/15;
8 GST=8+30/60+20/3600;//GST on GMM
9 i=delT*9.8565/3600;//error in time
10 pi=3.1415;
11 lst=GST+i;//LST on LMM
12 LST=19+52/60+16/3600;
13 RA=LST;
14 LMN=LST-lst;
15 i2=LMN*9.8565/3600;//error in LMN
```

```
LMN=LMN-i2;
OSM=10+55/60+30/3600;//Observed mean time
i3=1/60+25/3600;//error in observed time
OSM=OSM-i3;
LMT=OSM+4/15+21/60/15;
I=LMN-LMT;//interval
i4=1.21/3600;//error in interval
I=I+i4;
H=I;//hour angle
delta=6+15/60+02/3600;//deflection
alpha=39+20/60+30/3600;//altitude
theta=56+54/60+30/3600;//longitude
B=cos(delta*pi/180)*cos(theta*pi/180)/cos(alpha*pi
   /180);
m=225*H^2*3600^2/2/206265;
lat=alpha+m*B/3600;
lat=90-lat+6+15/60+02/3600;
disp(B,"latitude of star observed:");
a=modulo(lat*3600,60);
printf("seconds %.2f",a);
b=modulo(lat*3600-a,3600)/60;
printf("  minutes %i",b);
c=(lat*3600-b*60-a)/3600;
printf("  degrees %i",c);
```

Chapter 2

Photogrammetric Surveying

Scilab code Exa 2.1 Azimuth of camera

```
1
2
3 //exapple 2.1
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14259;
7 f=120.80//focal length
8 a=-35.52//elevation of A
9 b=8.48//elevation of B
10 c=48.26//elevation of C
11 alphaa=atan(a/f);
12 alphab=atan(b/f);
13 alphac=atan(c/f);
14 phi=(354+30/60)*pi/180;//azimuth of camera
15 phia=phi-alphaa-360*pi/180;//azimuth of a
16 disp(phia/pi*180,"azimuth of a in (degrees)");
17 phib=phia+alphab;//azimuth of b
18 disp(phib/pi*180,"azimuth of b in (degrees)");
19 phic=phia+alphac;//azimuth of c
20 disp(phic/pi*180,"azimuth of c in (degrees)");
21 clear
```

Scilab code Exa 2.2 distance of points

```
1
2
3 //exapple 2.2
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14259;
7 f=150;//focal length of camera
8 ap=20.2//elevation of a from p
9 aa1=16.4;//ditance to the right
10 aq=35.2//elevation of a from q
11 PQ=100;//distance of PQ
12 RL=126.845;//reduced level of instrument
13 alphap=atan(ap/f);
14 alphaq=atan(aq/f);
15 P=pi/3-alphap;//angle P
16 Q=40*pi/180-alphaq;//angle Q
17 A=pi-P-Q;//angle A;
18 AP=PQ*sin(Q)/sin(A);
19 disp(AP,"distance of AP in (m):");
20 AQ=PQ*sin(P)/sin(A);
21 disp(AQ,"distance of AQ in (m):");
22 Pa1=sqrt(ap^2+f^2);
23 AA1=aa1*AP/Pa1;
24 RLa=RL+AA1;//reduced level of A
25 disp(RLa,"reduced level of A in (M)")
```

Scilab code Exa 2.3 distance of points

```
1
```

```
2
3 //exapple 2.3
4 clc; funcprot(0);
5 // Initialization of Variable
6 pi=3.14259;
7 theta=(44+30/60)*pi/180;//angle b/w two points
8 x1=68.24;//distance of first point
9 x2=58.48;//distance of 2nd point
10 f=(x1+x2)/tan(theta)/2+sqrt((x1+x2)^2/4/(tan(theta))
     ^2+x1*x2);
11 disp(f,"focal length of lens in (mm):");
```

Scilab code Exa 2.4 representative fraction

```
1
2
3 //exapple 2.4
4 clc; funcprot(0);
5 // Initialization of Variable
6 //part1
7 pi=3.14259;
8 H=1200;//altitude
9 h=80;//elevation of hill
10 f=15/100;
11 R80=f/(H-h);
12 disp(R80,"representative fraction of hill is (times)
     :");
13 //part2
14 h=300;//elevation of hill
15 R300=f/(H-h);
16 disp(R300,"representative fraction of hill is (times
     ):");
```

Scilab code Exa 2.5 altitude of point

```
1
2
3 //exapple 2.5
4 clc; funcprot(0);
5 // Initialization of Variable
6 R=1/8000;
7 h=1500;
8 f=20/100;
9 H=h+f/R;
10 disp(H,"height above sea level in (m):");
```

Scilab code Exa 2.6 representative fraction

```
1
2
3 //exapple 2.6
4 clc; funcprot(0);
5 // Initialization of Variable
6 h=500;//elevaton of point
7 f=20/100;//focal length
8 v=8.65/100;//vertical distance of photograph
9 ho=2000;//horizontal distance of photograph
10 R=v/ho;//representative fraction
11 H=h+f/R;
12 disp(H,"height above sea level in (m):");
13 h=800;
14 S=(H-h)/f/100;//scale of photograph
15 disp(S,"1cm in photograph represents metres:")
16 clear
```

Scilab code Exa 2.7 altitude of point

```
1
2
3 //exapple 2.7
4 clc; funcprot(0);
5 // Initialization of Variable
6 m=1/50000;//map scale
7 pd=10.16;//photo distance
8 md=2.54;//map distance
9 R=pd/md*m;//representative fraction
10 f=16/100;
11 h=200;
12 H=h+f/R;
13 disp(H,"height above sea level in (m):");
```

Scilab code Exa 2.8 distance of points

```
1
2
3 //exapple 2.8
4 clc; funcprot(0);
5 // Initialization of Variable
6 f=20//focal length
7 xa=2.65;//x coordinate of a
8 xb=-1.92;//x coordinate of b
9 ya=1.36;//y coordinate of a
10 yb=3.65;//y coordinate of b
11 H=2500;
12 ha=500;//elevation of a
13 hb=300;//elevation of b
14 Xa=(H-ha)/f*xa;
15 Xb=(H-hb)/f*xb;
16 Ya=(H-ha)/f*ya;
17 Yb=(H-hb)/f*yb;
18 AB=sqrt((Xa-Xb)^2+(Ya-Yb)^2);
19 disp(AB,"distance between A & B in (m):")
```

```
20 clear
```

Scilab code Exa 2.9 altitude of a point

```
1
2
3 //exapple 2.9
4 clc; funcprot(0);
5 // Initialization of Variable
6 f=20//focal length
7 xa=2.65;//x coordinate of a
8 xb=-1.92;//x coordinate of b
9 ya=1.36;//y coordinate of a
10 yb=3.65;//y coordinate of b
11 ha=500;//elevation of a
12 hb=300;//elevation of b
13 ABg=545;
14 ab=5.112;
15 hab=ha/2+hb/2;
16 Happ=hab+ABg*f/ab
17
18 Xa=(Happ-ha)/f*xa;
19 Xb=(Happ-hb)/f*xb;
20 Ya=(Happ-ha)/f*ya;
21 Yb=(Happ-hb)/f*yb;
22 AB=sqrt((Xa-Xb)^2+(Ya-Yb)^2);
23 Hact=ABg/AB*(Happ-hab)+hab;
24 disp(Hact,"actual flying height of A & B in (m):");
```

Scilab code Exa 2.10 relief displacement

```
1
2
```

```
3 //exapple 2.10
4 clc; funcprot(0);
5 // Initialization of Variable
6 f=20/100;
7 Sd=1/10000;
8 h=250;//elevation
9 r=6.44;
10 H=f/Sd;
11 d=r*h/H;
12 disp(d,"relief displacement of the point in (cm)")
```

Scilab code Exa 2.11 relief displacement

```
1
2
3 //exapple 2.11
4 clc; funcprot(0);
5 // Initialization of Variable
6 h=50;//elevation
7 H=2500-1250;
8 r=6.35;
9 d=r*h/H;
10 disp(d,"relief displacement of the point in (cm)")
```

Scilab code Exa 2.12 relief displacement

```
1
2
3 //exapple 2.12
4 clc; funcprot(0);
5 // Initialization of Variable
6 f=20/100;//focal length
7 l=250;//length of line
```

```
lp=8.5/100;//length of line in photograph
H=l*f/lp;//height of camera above datum
r=6.46;//distace of image of top of the tower
d=0.46;//releif displacement
h=d*H/r;
disp(h,"height of tower above its base in (m)")
```

Scilab code Exa 2.13 flight plan

```

//exapple 2.13
clc; funcprot(0);
// Initialization of Variable
l=20/100;//length of photograph
w=20/100;//breadth of photograph
Pl=0.6;//longitudinal lap
Pw=0.3;//side lap
s=100*20;
L=(1-Pl)*s;
W=(1-Pw)*s;
Ar=L*W/1000/1000;
N=100/Ar;
A=round(N);
if N-A<0 then
   disp(A,"no. of photographs to be taken");
else
    disp(A+1,"no. of photographs to be taken");
end
```

Scilab code Exa 2.14 flight plan

```

```

```
2
3 //exapple 2.14
4 clc; funcprot(0);
5 // Initialization of Variable
6 Pl=0.6;//longitudinal lap
7 Pw=0.3;//side lap
8 L1=10000;
9 s=100*20;
10 L2=L1;
11 N1=L1/((1-Pl)*s)+1;
12 A1=round(N1);
13 if N1-A1<0 then
14    N1=A1
15 else
16     N1=A1+1;
17 end
18 N2=L2/((1-Pw)*s)+1;
19 A2=round(N2);
20 if N2-A2<0 then
21    N2=A2
22 else
23     N2=A2+1;
24 end
25 N=N1*N2;
26 disp(N,"no. of photographs to be taken");
27 clear
```

Scilab code Exa 2.15 photograph numbers

```
1
2
3 //exapple 2.15
4 clc; funcprot(0);
5 // Initialization of Variable
6 Pl=0.6;//longitudinal lap
```

```
Pw=0.3;//side lap
L1=12500;
s=100*20;
L2=8000;
N1=L1/((1-Pl)*s)+1;
A1=round(N1);
if N1-A1<0 then
   N1=A1
else
    N1=A1+1;
end
N2=L2/((1-Pw)*s)+1;
A2=round(N2);
if N2-A2<0 then
   N2=A2
else
    N2=A2+1;
end
N=N1*N2;
disp(N,"no. of photographs to be taken");
```

Scilab code Exa 2.16 flight planning

```


//exapple 2.16
clc; funcprot(0);
// Initialization of Variable
//part1
f=30/100;//focal length
h=400;//elevation of datum
r=12000;//ratio
s=120*20;
L2=24000;
L1=30000;
```

```
Pl=0.6;//longitudinal lap
Pw=0.3;//side lap
H=h+r*f;
disp(H,"height above datum in (m):");
//part2
W=(1-Pw)*s;
disp(W,"ground width covered in each photograph (m):
   ")
//part3
N2=L2/((1-Pw)*s)+1;
A2=round(N2);
if N2-A2<0 then
   N2=A2
else
    N2=A2+1;
end
disp(N2,"no. of flights required")
//part4
Asf=L2/(N2-1);//actual spacing between flights
//part5
Sfl=Asf/600;//spacing of flight lines
//part6
gd=(1-Pl)*s;//ground distance
//part7
Ei=gd/55.5;//exposure interval
Ei=round(Ei);
//part8
Ags=55.56*Ei;//adgusted ground distance
//part9
N1=L1/Ags+1;
A1=round(N1);
if N1-A1<0 then
   N1=A1
else
    N1=A1+1;
end
N=N1*N2;
disp(N,"no. of photographs to be taken");
```

Scilab code Exa 2.17 parallax of a point

```
1
2
3 //exapple 2.17
4 clc; funcprot(0);
5 // Initialization of Variable
6 f=150/1000;//focal length
7 r=20000;//ratio
8 Pl=0.6;//longitudinal lap
9 l=23/100;//length
10 w=23/100;//width
11 B=(1-Pl)*l*r;//base length
12 H=f*r;
13 h=0;
14 dh=(H-h)^2/B/f*0.1/1000;
15 disp(dh,"error in height in (m):")
```

Scilab code Exa 2.18 parallax

```
1
2
3 //exapple 2.18
4 clc; funcprot(0);
5 // Initialization of Variable
6 H=600;
7 f=150/1000;
8 s=H/f;
9 b=6.375/100;
10 h1=0;
11 h2=120;//height of chimney
```

```
12 B=s*b;//datum elevation
13 p1=B*f*1000/(H-h1);
14 p2=B*f*1000/(H-h2);
15 delp=p2-p1;
16 delh=H*delp/1000/(b+delp/1000);
17 disp(delh,"parallax height of the chimney in (m):");
18 clear
```

Scilab code Exa 2.19 parallax

```
1
2
3 //exapple 2.19
4 clc; funcprot(0);
5 // Initialization of Variable
6 B=200;
7 f=120;
8 p2=52.52;//parallax for top of pole
9 p1=48.27;//parallax for bottom of pole
10 delh=(p2-p1)/p2/p1*B*f;
11 disp(delh,"diference in elevation of two points in (
     m):")
```

Scilab code Exa 2.20 parallax

```
1
2
3 //exapple 2.20
4 clc; funcprot(0);
5 // Initialization of Variable
6 //part1
7 delp=1.48/1000;
8 H=5000;
```

```
h=500;
b=90/1000;//mean principal base
dh=(H-h)^2*delp/((H-h)*delp+b*H);
disp(dh,"difference in height between two points in (m):");
//part2
delp=15.5/1000;
dh=(H-h)^2*delp/((H-h)*delp+b*H);
disp(dh,"difference in height between two points in (m):");
```