Scilab Textbook Companion for
Discrete Mathematics
by S. Lipschutz, M. Lipson And V. H. Patil[1]

Created by
Neerja Bahuguna
B.Tech. (pursuing)
Computer Engineering
Echelon Institute of Technology, Faridabad
College Teacher
Prashant Gupta, E. I. T. ,Faridabad
Cross-Checked by
Prashant Dave, IIT Bombay

May 17, 2016

Book Description

Title: Discrete Mathematics

Author: S. Lipschutz, M. Lipson And V. H. Patil

Publisher: Tata McGraw - Hill Education

Edition: 3

Year: 2009

ISBN: 9780070669123

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

Set Theory

Scilab code Exa 1.8 inclusion exclusion principle

```
1  disp ('To find:number of mathematics students taking
       atleast one of the languages French(F),German(G)
       and Russian(R)')
2  F=65; //number of students studying French
3  G=45;  // number of students studying German
4  R=42;  //number of students studying Russian
5  FandG=20;  //number of students studying French and
       German
6  FandR=25;  //number of students studying French and
       Russian
7  GandR=15;   //number of students studying German and
       Russian
8  FandGandR=8; //number of students studying French,
       German and Russian
9   //By inclusion-exclusion principle
10   ForGorR=F+G+R-FandG-FandR-GandR+FandGandR;
11 disp (ForGorR, 'the number of students studying
       atleast one of the languages :')
```

Scilab code Exa 1.9 Inclusion exclusion principle

```
1  disp('In a college ,120 mathematics students can opt
       for either French(F),German(G) or Russian(R)')
2  n=120;   //total number of students
3  F=65;   //number of students studying French
4  G=45;   //number of students studying German
5  R=42;   //number of students studying Russian
6  FandG=20;   //number of students studying French and
       German
7  FandR=25;   //number of students studying French and
       Russian
8  GandR=15;   //number of students studying German and
       Russian
9  FandGandR=8;   //number of students studying French ,
       German and Russian
10 disp('using inclusion-exclusion principle:')
11 ForGorR=F+G+R-FandG-FandR-GandR+FandGandR;
12 disp(ForGorR,'number of students studying French or
       German or Russian')
13 FGnR=FandG-FandGandR;
14 disp(FGnR,'number of students studying French and
       German but not Russian')
15 FRnG=FandR-FandGandR ;
16 disp(FRnG,'number of students studying French and
       Russian but not German')
17 GRnF=GandR-FandGandR ;
18 disp(GRnF,'number of students studying German and
       Russian but not French')
19 OF=F-FGnR-FandGandR-FRnG ;
20 disp(OF,'number of students studying  Only French')
21 OG=G-FGnR-FandGandR-GRnF;
22 disp(OG,'number of students studying  Only German')
23 OR=R-FRnG-FandGandR-GRnF;
24 disp(OR,'number of students studying  Only Russian')
25 k=n-ForGorR;
26 disp(k,'number of students not studying any of the
       languages')
```

Scilab code Exa 1.13 Power sets

```
1 x=10; //number of members of set X
2 P=2^x //number of members of the power set of X
3 q=P-1;//x itself is not the proper subset.Hence it
     isn't counted
4 disp(q,'number of members of powerset P which are
     proper subsets of x are:')
```

Scilab code Exa 1.14 Power sets

```
1 A=[1,2,3,4,5];  //eatables for salad preparation 1=
     onion,2=tomato,3=carrot,4=cabbage,5=cucumber
2 p=length(A);   //total number of eatables available
3 n=2^p-1;  //no salad can be made without atleast one
      of the eatables.Hence null set isn't counted
4 disp(n,'number of different salads that can be
     prepared using the given eatables')
```

Scilab code Exa 1.18 Mathematical induction

```
1 U1=1;  //given
2 U2=5;  //given
3 P=[];
4 for i=1:2
5 P(i)=3^i-2^i;
6 disp(P(i))
7 end
8 disp('P(1)=U(1) and P(2)=U(2)');
```

```
9  disp ('hence Un=3^n-2^n for all n belonging to N');
```

Chapter 3

Functions and Algorithms

Scilab code Exa 3.8 Recursively defined functions

```
1  function[k]=fact(a)
2  k=-1;
3  if(a<0|a>200)
4  disp("Invalid");
5  break;
6    else
7  if(a==1|a==0)
8  k=1;
9    else
10 k=a*fact(a-1);
11 end
12 end
13 endfunction
14 a=4;
15 p=fact(a);
16 disp(p,'the value of 4! is')
```

Scilab code Exa 3.9 Cardinality

```
1  x =1;
2  y =2;
3  z =3;
4  A =[x ,y ,z ];
5  disp ( 'cardinality of set A is: ')
6  length (A)
7  B =[1 ,3 ,5 ,7 ,9]
8  disp ( 'cardinality of set B is: ')
9  length (B)
10
11 // 3.9 (b)
12 disp ( 'the set E has the following elements )
13 E =[2 ,4 ,6 %inf]  //set E is the set of all positive
       even numbers and N is the set of all natural
       numbers
14 disp ( 'function f:N to E is defined.So ,E has the same
        cardinality as N ')
15 disp ( 'set E is countably infinite: ')
16 for x =2:2:%inf
17 y =2* x ;
18 disp (y)
19 end
```

Scilab code Exa 3.10 Polynomial evaluation

```
1  x = poly (0 , 'x ');
2  p = 2*x^3-7*x^2+4*x-15;
3  disp (p , 'the polynomial is ')
4  k =horner (p ,5);
5  disp (k , 'value of the polynomial at x=5 is ')
```

Scilab code Exa 3.11 Greatest Common Divisor

```
1 V=int32([258,60]);
2 thegcd=gcd(V);
3 disp(thegcd,'the gcd of the two numbers 258 and 60
      is')
```

Chapter 5

Vectors and Matrices

Scilab code Exa 5.2 Vector operations

```
1  u=[2,3,-4];
2  v=[1,-5,8];
3  u+v
4  5*u
5  -v
6  2*u-3*v
7  u.*v;
8  k=sum(u.*v);
9  disp(k,'dot product of the two vectors')
10 l=norm(u);
11 disp(l,'norm or length of the vector u')
```

Scilab code Exa 5.3 Column vectors

```
1  u=[5,3,-4]'
2  v=[3,-1,-2]'
3  2*u-3*v
4  k=sum(u.*v);
```

```
5 disp(k,'The dot product of the two vectors u and v
     is :')
6 l=norm(u);
7 disp(l,'The length or norm of the vector u is:')
```

Scilab code Exa 5.5 Matrix addition and Scalar multiplication

```
1 A=[1,-2,3;0,4,5];
2 B=[4,6,8;1,-3,-7];
3 k=A+B;
4 disp(k,'The addition of the two matrices A and B is:
     ')
5 m=3*A;
6 disp(m,'The multiplication of a vector with a scalar
      is:')
7 p=2*A-3*B
```

Scilab code Exa 5.6 Matrix multiplication

```
1 a=[7,-4,5];
2 b=[3,2,-1]';
3 k=a*b;
4 disp(k,'product of a and b is;')
5 p=[6,-1,8,3];
6 q=[4,-9,-2,5]';
7 l=p*q;
8 disp(l,'product of p and q is:')
```

Scilab code Exa 5.7 Matrix multiplication

```
1  A =[1  3;2  -1];
2  B =[2  0  -4;5  -2  6];
3  A*B
4  A =[1  2;3  4]
5  B =[5  6;0  -2];
6  A*B
7  B*A
8  disp('matrix  mulitplication  is  not  commutative  since
        AB  may  not  be  equal  to  BA')
```

Scilab code Exa 5.8 Algebra of square matrices

```
1  A =[1  2;3  -4];
2  A2=A*A   //multiplying  A  by  itself
3  A3=A2*A
4  f =2*A2-3*A+5;
5  disp(f,'for  the  function  f(x)=2xˆ2-3x+5,f(A)  is:')
6  g=A2+3*A-10;
7  disp(g,'for  the  function  g(x)=xˆ2+3x-10,g(A)  is ')
```

Scilab code Exa 5.9 Invertible matrices

```
1  A =[1  0  2;2  -1  3;4  1  8];
2  B =[-11  2  2;-4  0  1;6  -1  -1];
3  A*B
4  disp('since  A*B  is  identity  matrix ,A  and  B  are
        invertible  and  inverse  of  each  other ')
```

Scilab code Exa 5.10 Determinants

```
1 A =[5 4;2 3];
2 det ( A );
3 disp ( det ( A ) ,'determinant of A')
4 B =[2 1; -4 6]
5 det ( B );
6 disp ( det ( B ) ,'determinant of B')
7 C =[2 1 3;4 6 -1;5 1 0]
8 disp ( det ( C ) ,'determinant of C')
```

Scilab code Exa 5.13 Matrix solution of a system of linear equations

```
1 A =[1 2 1;2 5 -1;3 -2 -1];        // left hand side of
      the system of equations
2 B =[3 -4 5]';                     // right hand side or
      the constants in the equations
3 X =[];
4 X = A \ B ;    // unique solution for the system of
      equations
5 x = X (1)
6 y = X (2)
7 z = X (3)
```

Scilab code Exa 5.14 Inverse of a square matrix

```
1 A =[1 0 2;2 -1 3;4 1 8];
2 P = rref ([ A , eye (3 ,3) ]) ;
3 disp ( P ,'canonical form of matrix A :')
4 disp ('left side of the matrix P is the identity
      matrix so the right side is the inverse of A')
5 inverseA = P (: ,4:6)
```

Chapter 6

Counting

Scilab code Exa 6.1 Sum rule principle

```
1  M=8;      //number of male professors teaching
      calculus
2  F=5;      //number of female professors teaching
      calculus
3  T=M+F ;
4  disp(T,'number of ways a student can choose a
      calculus professor')
5
6  E=[2,3,5,7];   //event of choosing a prime number
      less than 10
7  F=[2,4,6,8];   //event of choosing an even number
      less than 10
8  G=intersect(E,F);  //event of getting an even and
      prime number
9  H=length(E)+length(F)-length(G);
10 disp(H,'event of getting an even or a prime number')
11
12 E=[11,13,17,19]; //event of choosing a prime number
      between 10 and 20
13 F=[12,14,16,18];  //event of choosing an even number
      between 10 and 20
```

```
14  G=union(E,F);      //event of choosing a number which
        is prime or even
15  k=length(G);
16  disp(k,'number of ways of choosing a number which is
         prime or even ')
```

Scilab code Exa 6.2 Product rule principle

```
1  disp('a license plate contains two letters followed
      by three digits where first digit can not be zero
      ')
2  n=26; //number of english letters
3  n*n;  //number of ways of choosing two letters in
      the license plate
4  p=10; //number of digits (0-9)
5  (p-1)*p*p; //number of ways to select the three
      digits with the first digit not being zero
6  k=n*n*(p-1)*p*p;
7  disp(k,'total number of license plates that can be
      printed ')
8
9  disp('a president ,a secretary and a treasurer has
      to be elected in an orga-nisation of 26 members.
      No person is elected to more than one postion ')
10  t=26;  //total number of members in the organisation
11  j=t*(t-1)*(t-2);
12  disp(j,'number of ways to elect the three officers (
      president ,secretary ,treasurer ')
```

Scilab code Exa 6.3 Factorial notation

```
1  disp('To find: factorial of a 6')
2  facto2=2*1;
```

```
3  facto3=3*facto2
4  facto4=3*facto3
5  facto4=4*facto3
6  facto5=5*facto4
7  facto6=6*facto5
8
9  k=8*7*factorial(6)/factorial(6);
10 disp(k,'value of 8!/6! is:')
11 j=12*11*10*factorial(9)/factorial(9);
12 disp(j,'value of 12!/9! is:')
```

Scilab code Exa 6.4 Binomial coefficients

```
1  function [k]=func1(n,r)   //calculating binomial
       coefficient
2  k=factorial(n)/(factorial(r)*factorial(n-r));
3  endfunction
4  func1(8,2)
5  func1(9,4)
6  func1(12,5)
7  func1(10,3)
8  func1(13,1)
9
10 p = factorial(10)/(factorial(10-7)*factorial(7));
       //calculating 10C7
11 q= factorial(10)/(factorial(10-3)*factorial(3))   //
       calculating 10C3
12 disp(p,'value of 10C7 is')
13 //10-7=3 so 10C7 can also be computed as 10C3
14 //both p and q have same values but second method
       saves time and space
```

Scilab code Exa 6.5 Permutations

```
1 disp('finding the number of three-letter words using
        only the given six letters(A,B,C,D,E,F) without
      repetition')
2 n=6;  //total number of letters
3 l1=n;  //number of ways in which first letter of the
        word can be chosen
4 l2=n-1; //number of ways in which second letter of
      the word can be chosen
5 l3= n-2; //number of ways in which third letter can
      be chosen
6 k=l1*l2*l3;
7 disp(k,'number of three-letter words  possible')
```

Scilab code Exa 6.6 Permutations with repetitions

```
1 function [k]= funct1(n,p,q)
2 k= factorial(n)/(factorial(p)*factorial(q));
3 endfunction
4 k=funct1(7,3,2)  //in "BENZENE" three letters are
      alike(the three Es) and two are alike (the two Ns
      )
5 disp(k,'The number of seven-letter words that can be
       formed using letters of the word BENZENE')
6
7 disp('a set of 4 indistinguishable red coloured
      flags, 3 indistinguishable white flags and a blue
       flag is given')
8 j=funct1(8,4,3);
9 disp(j,'number of different signals ,each consisting
      of eight flags')
```

Scilab code Exa 6.7 Combinations

```
1 disp('four objects are given (a,b,c,d) and three are
      taken at a time')
2 combinations = factorial(4)/(factorial(4-3)*
     factorial(3));
3 disp(combinations,'number of combinations of the
     four objects given')
4 k=factorial(3);  //number of permutations of objects
      in a combination
5 permutations = combinations*k;
6 disp(permutations,'total number of permuatations for
      the problem')
```

Scilab code Exa 6.8 Combinations

```
1 function [k]= myfunc(n,r)
2 k=factorial(n)/(factorial(n-r)*factorial(r));
3 endfunction
4 k=myfunc(8,3);
5 disp(k,'the number of committees of three that can
     be formed out of eight people')
6
7 cows=myfunc(6,3)  //number of ways that a farmer can
      choose 3 cows out of 6 cows
8 pigs=myfunc(5,2)  //number of ways that a farmer can
      choose 2 pigs out of 5 pigs
9 hens=myfunc(8,4)  //number of ways that a farmer can
      choose 4 hens out of 8 hens
10 p=cows*pigs*hens;
11 disp(p,'total number of ways that a farmer can
     choose all these animals')
```

Scilab code Exa 6.9 Combinations with repetitions

```
1 //each solution  to the equation can be viewed as a
     combination of objects
2 r=18;   //number of objects
3 M=3;    //kinds of object
4 m=factorial(r+(M-1))/(factorial(r+(M-1)-(M-1))*
     factorial(M-1));
5 disp(m,'number of non negative integer solutions of
     the given equation x+y+z=18')
```

Scilab code Exa 6.14 Ordered partitions

```
1 c1=3;  //number of toys that the youngest child
     should get
2 c2=2;   //number of toys that  the third child should
      get
3 c3=2;   //number of toys that the second child should
      get
4 c4=2;    //number of toys that the eldest son should
     get
5 m=factorial(9)/(factorial(3)*factorial(2)*factorial
     (2)*factorial(2));
6 disp(m,'number of ways nine toys can be divided
     between four children with the youngest son
     getting 3 toys and others getting 2 each')
```

Scilab code Exa 6.15 Unordered partitions

```
1 p=12;   //total number of students
2 t=3;    //number of teams or partition
3 disp('each partition of the students can be arranged
      in 3! ways as an ordered partition')
4 r=factorial(12)/(factorial(4)*factorial(4)*factorial
     (4))   //number of ordered partitions
```

```
5  m=r/factorial(t);  //number of unordered partitions
6  disp(m,'number of ways that 12 students can be
       partitioned into three teams so that each team
       consists of 4 students ')
```

Scilab code Exa 6.16 Inclusion exclusion principle revisited

```
1  U=1000;  //number of elements in the set of positive
        integers not exceeding 1000
2  A=U/3;    //number of elements in the subset of
      integers divisible by 3
3  B=U/5;    //number of elements in the subset of
      integers divisible by 5
4  C=U/7;    //number of elements in the subset of
      integers divisible by 7
5  AandB=floor(U/(3*5))   //number of elements in the
      subset containing numbers divisible by both 3 and
       5
6  AandC=floor(U/(3*7))   //number of elements in the
      subset containing numbers divisible by both 3 and
       7
7  BandC=floor(U/(5*7))   //number of elements in the
      subset containing numbers divisible by both 5 and
       7
8  AandBandC=floor(U/(3*5*7))  //number of elements in
      the subset containing numbers divisible by 3,5
      and 7
9  s=U-(A+B+C)+(AandB+AandC+BandC)-(AandBandC);  // By
      inclusion-exclusion principle
10 S=round(s);
11 disp(S,'The number of integers in the set U, which
      are not divisible by 3,5 and 7 is ')
```

Chapter 7

Probability Theory

Scilab code Exa 7.1 Sample space and events

```
1  S =[1 ,2 ,3 ,4 ,5 ,6];          // sample  space  for  the
       rolling  of  a  die
2  A =[2 ,4 ,6];                  // event  that  an  even  number
       occurs
3  B =[1 ,3 ,5];                  // event  that  an  odd  number
       occurs
4  C =[2 ,3 ,5];                  // event  that  a  prime  number
       occurs
5  disp ( union (A ,C ) , 'sample  space  for  the  event  that  an
       even  or  a  prime  number  occurs ')
6  disp ( intersect (B ,C ) , 'sample  space  for  the  event  that
        an  odd  prime  number  occurs ')
7  disp ( setdiff (S ,C ) , 'sample  space  for  the  event  that  a
        prime  number  does  not  occur ')       // It  is  the
       complement  of  the  set  C.
8  intersect (A ,B )       // It  is  a  null  set  or  null  vector
       since  there  can 't  occur  an  even  and  an  odd
       number  simultaneously
9
10 H =0;                  // " head "  face  of  a  coin
11 T =1;                  // " tail "  face  of  a  coin
```

```
12  S =[ " 000 " , " 001 " , " 010 " , " 011 " , " 100 " , " 101 " , " 110 " , " 111 " ]
        ;        // sample  space  for  the  toss  of  a  coin  three
        times
13  A =[ " 000 " , " 001 " , " 100 " ];           // event  that  two  more  or
        more  heads  appear  consecutively
14  B =[ " 000 " , " 111 " ];                 // event  that  all  tosses
        are  the  same
15  disp ( intersect (A , B ) , ' sample  space  for  the  event  in
        which  only  heads  appear ')
16
17  disp ( ' Experiment : tossing  a  coin  until  a  head  appears
         and  then  counting  the  number  of  times  the  coin
        is  tossed ')
18  S =[1 ,2 ,3 ,4 ,5 ,% inf ]          // The  sample  space  has
        infinite  elements  in  it
19  disp ( " Since  every  positive  integer  is  an  element  of
        S , the  sample  space  is  infinite " )
```

Scilab code Exa 7.2 Finite probability spaces

```
1  disp ( ' Experiment : three  coins  are  tossed  and  the
        number  of  heads  are  observed ')
2  S =[0 ,1 ,2 ,3 ];    // the  sample  space  for  the  experiment
         where  0  implies  no  heads ,1  implies  only  one  head
         out  of  the  three  coins  and  so  on
3  disp ( " the  probability  space  is  as  follows  " )
4  P0 =1/8;   // probability  of  getting  no  head  on  any  of
        the  coins  i . e  TTT
5  P1 =3/8;   // probability  of  getting  only  one  head  on
        any  of  the  coins ,  out  of  the  three  coins  i . e  HTT,
        THT , TTH
6  P2 =3/8;   // probability  of  getting  two  heads ,  out  of
        the  three  coins  i . e  THH , HTH , HHT
7  P3 =1/8;     // probability  of  getting  all  the  three
        heads  i . e  HHH
```

```
8 disp("A is the event that atleast one head appears
      and B is the event that all heads or all tails
      appear ")
9 A=[1,2,3];    // HHH,HHT,HTH,HTT,THH,THT,TTH
10 B=[0,3];     //HHH,TTT
11 PA=P1+P2+P3;
12 disp(PA,'probability of occurrence of event A')
13 PB=P0+P3;
14 disp(PB,'probability of occurrence of event B')
```

Scilab code Exa 7.3 Equiprobable spaces

```
1 disp("Experiment: a card is selected from a deck of
      52 cards ")
2 disp("A is the event of the selected card being a
      spade ")
3 disp("B is the event of the selected card being a
      face card ")
4 t=52 ;        //the total number of cards
5 s=13;         //number of spades
6 PA= s/t;
7 disp(PA,'probability of selecting a spade')
8 f=12;         //number of face cards(jack,queen,king)
9 PB=f/t;
10 disp(PB,'probability of selecting a face card')
11 sf=3;        //number of spade face cards
12 Psf=sf/t;
13 disp(Psf,"probability of selecting a spade face card
      is:")
```

Scilab code Exa 7.4 Addition principle

```
1 disp("Experiment: selection of a student out of 100
      students ")
2 M=30;   //no of students taking mathematics
3 C=20;   //no of students taking chemistry
4 T=100;  //total no. of students
5 PM = M/T  //probability of the selected student
     taking mathematics
6 PC = C/T  //probability of the selected student
     taking chemistry
7 MnC=10;  //no of students taking mathematics and
     chemistry
8 PMnC = MnC/T   //probability of the selected student
      taking mathematics and chemistry both
9 PMorC = PM+PC-PMnC;
10 disp(PMorC,'probability of the selected student
     taking mathematics or chemistry')
```

Scilab code Exa 7.5 Conditional probability

```
1
2
3 //EXAMPLE 7.5 (a)
4
5
6  disp(" Experiment: A die is tossed and the outcomes
       are observed");
7
8
9  disp("To find: probability (PM) of an event that
     one of the dice is 2 if the sum is 6");
10
11
12 E=[" (1 ,5)"," (2 ,4)"," (3 ,3)"," (4 ,2)"," (5 ,1)"]     //
     event that the sum of the two numbers on the two
     dice is 6
```

```
13

14

15  A=[" (2 ,1) " ," (2 ,2) " ," (2 ,3) " ," (2 ,4) " ," (2 ,5) " ," (2 ,6) " ,"
        (1 ,2) " ," (3 ,2) " ," (4 ,2) " ," (5 ,2) " ," (6 ,2) "]    // event
        that 2 appears on atleast one die

16

17

18  B= intersect (A ,E)    // possible combination of
        numbers on two die such that their sum is 6 and 2
         appears atleast on one die

19

20

21  PM =2/5   // since E has 5 elements and B has 2
        elements

22

23

24

25

26

27  //EXAMPLE 7.5( b )

28

29  disp ("A couple has two children ");

30

31

32  b =1;    // boy child

33

34  g =2;    // girl child

35

36  S=[11 ,12 ,21 ,22] ;      // sample space where 11 implies
         both children being boys ,12 implies first child
        being a boy and the second child being a girl
                                       and so on

37

38  disp ("To find : probability (PM) that both children
        are boys  ");

39

40

41
```

```
42  //7.5(b).i
43
44  L=S(:,1:3)        //reduced sample space if it is known
         that one of the children is a boy
45
46
47  PM=1/length(L)
48
49
50  //7.5(b).ii
51
52  R=S(:,1:2)      //reduced sample space if it is known
        that the older child is a boy
53
54
55  PM=1/length(R)
```

Scilab code Exa 7.6 Multiplication theorem for conditional probability

```
1  disp("A bag contains 12 items of which four are
        defective.Three items are drawn at random,one
        after the other");
2  s=12;    //total itmes in the bag
3  d=4;   //defective items in the bag
4  Pf=(s-d)/s ;   //probability that the first item
        drawn is  non defective
5  Pe=Pf*[(s-d-1)/(s-1)]*[(s-d-2)/(s-2)];
6  disp(Pe,'probability that all three items are non
        defective')
7  //after the first item is chosen,the second item is
        to be chosen from 1 less than the original number
         of items in the box and similarly the number of
        non defective items gets decreased by 1.Similarly
         ,for the third draw of item from the box
```

Scilab code Exa 7.7 Independent events

```scilab
1  H=1;     //heads of a coin
2  T=2;     //tails of the coin
3  S=[111,112,121,122,211,212,221,222]  //sample space
      for the toss of a coin three times. 111 implies
      heads all three times,112 implies heads on first
      two tosses and tails on the third toss
4  A=[111,112,121,122];  //event that first toss is
      heads
5  B=[111,112,211,212];  //event that second toss is
      heads
6  C=[112,211];            //event that exactly two heads
      appear in a row
7  PA=length(A)/length(S);
8  disp(PA,'probability of A is')
9  PB=length(B)/length(S);
10 disp(PB,'probability of B is')
11 PC=length(C)/length(S);
12 disp(PC,'probability of C is')
13 AnB=intersect(A,B)
14 AnC=intersect(A,C)
15 BnC=intersect(B,C)
16 PAnB= length(AnB)/length(S);
17 disp(PAnB,'probability of the event AnB')
18 PAnC= length(AnC)/length(S);
19 disp(PAnC,'probability of the event AnC')
20 PBnC= length(BnC)/length(S);
21 disp(PBnC,'probability of the event BnC')
22 if((PA*PB)==PAnB),
23  disp("A and B are independent")
24 else
25  disp("A and B are dependent")
26 end
```

```
27  if((PA*PC)==PAnC),
28     disp("A and C are independent")
29  else
30   disp("A and C are dependent")
31  end
32  if((PB*PC)==PBnC),
33     disp("B and C are independent")
34  else
35     disp("B and C are dependent")
36  end
```

Scilab code Exa 7.8 Independent events

```
1  disp("Experiment: A and B both shoot at a target")
2  PA=1/4;   //given probability of A hitting the target
3  PB=2/5;   //given probability of B hitting the target
4  disp("A and B are independent events so PA*PB will
       be equal to probability of the event of A and B
       both hitting the target i.e PAnB")
5  PAnB=PA*PB;
6  PAorB=PA+PB-PAnB;
7  disp(PAorB,'probability of atleast one of them
       hitting the target ')
```

Scilab code Exa 7.9 Independent repeated trials

```
1  disp("Experiment: Three horses race together twice")
2  Ph1=1/2;   //probability of first horse winning the
       race
3  Ph2=1/3;    //probability of second horse winning the
        race
4  Ph3=1/6;    //probability of third horse winning the
       race
```

```
5  S=[11 ,12 ,13 ,21 ,22 ,23 ,31 ,32 ,33]   // sample  space  where
       11  implies  first  horse  winning  the  first  and
       second  race  both ,12  implies  first  horse  winning
       the  first  race  and  second  horse  winning  the
       second  race  and  so  on
6  P11= Ph1 * Ph1   // probability  of  first  horse  winning
       both  races
7  P12= Ph1 * Ph2   // probability  of  first  horse  winning
       the  first  race  and  second  horse  winning  the
       second  race
8  P13= Ph1 * Ph3   // probability  of  first  horse  winning
       the  first  race  and  third  horse  winning  the  second
        race
9  P21= Ph2 * Ph1   // probability  of  second  horse  winning
       the  first  race  and  first  horse  winning  the  second
        race
10 P22= Ph2 * Ph2   // probability  of  second  horse  winning
       both  the  races
11 P23= Ph2 * Ph3   // probability  of  second  horse  winning
       the  first  race  and  third  horse  winning  the  second
        race
12 P31= Ph3 * Ph1   // probability  of  third  horse  winning
       the  first  race  and  first  horse  winning  the  second
        race
13 P32= Ph3 * Ph2   // probability  of  third  horse  winning
       the  first  race  and  second  horse  winning  the
       second  race
14 P33= Ph3 * Ph3   // probability  of  third  horse  winning
       both  the  races
15 disp (P31 , ' probability  of  third  horse  winning  the
       first  race  and  first  horse  winning  the  second
       race  is ')
```

Scilab code Exa 7.10 Repeated trials with two outcomes

```
1  n=6;      //number of times a fair coin is tossed and
       getting a heads is a success
2  p=1/2;   //probability of getting a heads
3  q=1/2 ;  //probability of not getting a heads
4  P2=(factorial(6)/(factorial(6-2)*factorial(2)))*p^2*
       q^(6-2);
5  disp(P2,'probability of getting exactly two heads (i
       .e k=2)')
6
7  P4=(factorial(6)/(factorial(6-4)*factorial(4)))*p^4*
       q^(6-4); //probabilty of getting four heads
8  P5=(factorial(6)/(factorial(6-5)*factorial(5)))*p^5*
       q^(6-5);  //probabilty of getting five heads
9  P6=(factorial(6)/(factorial(6-6)*factorial(6)))*p^6*
       q^(6-6);  //probabilty of getting five heads
10  PA=P4+P5+P6 ;
11  disp(PA,'probability of getting atleast four heads(i
       .e k=4,5 or 6)')
12
13  Pn=q^6            //probability of getting no heads
14  Pm=1-Pn;
15  disp(Pm,'probability of getting one or more heads')
```

Scilab code Exa 7.12 Random variables

```
1  disp("A box contains 12 items of which three are
       defective")
2  disp("A sample of three items is selected from the
       box")
3  s=factorial(12)/(factorial(12-3)*factorial(3));
4  disp(s,'number of elements in the sample space where
       samples are of size 3')
5  //X denotes the number of defective items in the
       sample
6  x=[0,1,2,3];   //range space of the random variable X
```

Scilab code Exa 7.13 Probability distribution of a random variable

```
1  r=[1,2,3,4,5,6,5,4,3,2,1];
2  //number of outcomes whose sum is
       2,3,4,5,6,7,8,9,10,11,12 respectively such that
       there is only 1 outcome i.e (1,1) whose sum is 2,
       two outcomes (1,2) and (2,1) whose sum is 3 and
       so on
3  t=36;                      //total number of
       elements in the sample space of the experiment of
        tossing a pair of dice
4  for i=1:11;
5  p=r(i)/t;
6  disp(p)
7  end
8  0.0277778             //probability of getting a sum
          of 2
9  0.0555556              //probability of getting a
       sum of 3
10 0.0833333             //probability of getting a sum
          of 4
11 0.1111111             //probability of getting a sum
          of 5
12 0.1388889             //probability of getting a sum
          of 6
13 0.1666667             //probability of getting a sum
          of 7
14 0.1388889             //probability of getting a sum
          of 8
15 0.1111111             //probability of getting a sum
          of 9
16 0.0833333             //probability of getting a sum
          of 10
17 0.0555556             //probability of getting a sum
```

```
        of 11
18 0.0277778              //probability of getting a sum
       of 12
19 x=[2,3,4,5,6,7,8,9,10,11,12];        //range space of
       random variable X which assigns to each point in
       sample space the sum of the numbers
20 D=[ 2,3,4,5,6,7,8,9,10,11,12;   0.0277778, 0.0555556
       , 0.0833333,  0.1111111, 0.1388889 ,0.1666667,
       0.1388889 ,0.1111111, 0.0833333, 0.0555556,
       0.0277778];
21 disp(D,'distribution table of X where first row
       gives the range space and second row gives the
       respective probabilities is as follows')
```

Scilab code Exa 7.14 Probability distribution of a random variable

```
1 disp("a box contains 12 items of which three are
       defective")
2 disp("A sample of three items is selected from the
       box")
3 r=factorial(9)/(factorial(9-3)*factorial(3))  //
       number of samples of size 3 with no defective
       items
4 t=220;                                          //
       number of different samples of size 3 i.e the
       number of elements in the sample space
5 P0=r/t                                          //
       probability of getting no defective item
6 r1=3*(factorial(9)/(factorial(9-2)*factorial(2)))
           //number of samples of size 3 getting 1
       defective item
7 P1=r1/t

       //probability of getting 1 defective item
8 r2=9*(factorial(3)/(factorial(3-2)*factorial(2)))
```

```
                //number  of  samples  of  size  3  getting  2
      defective  item
 9  P2=r2/t
                                                              //
      probability  of  getting  2  defective  item
10  r3=1;                          //number  of  samples  of
      size  3  getting  3  defective  item
11  P3=r3/t                        //probability  of  getting  3
       defective  item
12  x=[0,1,2,3];
13  p=[P0,P1,P2,P3];
14  D=[0,1,2,3;P0,P1,P2,P3];
15  disp(D,'distribution  table  for  random  variable  X  the
       upper  row  being  values  of  X')
```

Scilab code Exa 7.15 Expectation of a random variable

```
 1  disp("A  fair  coin  is  tossed  six  times");
 2  x=[0,1,2,3,4,5,6];    //number  of  heads  which  can
      occur
 3  p=[1/64,6/64,15/64,20/64,15/64,6/64,1/64];   //
      probability  of  occurring  of  heads  where  1/64  is
      probability  for  occurrence  of  a  single  head,6/64
      that  of  occurrence  of  two  heads  and  so  on.
 4  r=0;
 5  for  i=1:7;
 6  r  =  r  +  (x(i)*p(i));
 7  end
 8  disp(r,'mean  or  expected  number  of  heads  are')
 9
10  disp("X  is  a  random  variable  which  gives  possible
      number  of  defective  items  in  a  sample  of  size  3")
      ;
11  //Box  contains  12  items  of  which  three  are  defective
12  x=[0,1,2,3];    //possible  number  of  defective  items
```

37

```
      in  a  smaple  of  size  3
13  p =[84/220 ,108/220 ,27/220 ,1/220];    // p r o b a b i l i t y  of
      occurrence  of  each  number  in  x  respectively  where
       84/220  is  the  probability  for  getting  no
      defective  item ,108/220  is  that  of  getting  1
      defective  item  and  so  on.
14   r =0;
15  for  i =1:4;
16  r  =  r  +  (x(i)*p(i));
17  end
18  disp (r, 'expected  number  of  defective  items  in  a
      sample  of  size  3  are ')
19
20  Ph1 =1/2;      // p r o b a b i l i t y  of  winning  the  race  by
      first  horse
21  Ph2 =1/3;      // p r o b a b i l i t y  of  winning  the  race  by
      second  horse
22  Ph3 =1/6;      // p r o b a b i l i t y  of  winning  the  race  by
      third  horse
23  //X  is  the  payoff  function  for  the  winning  horse
24  X1 =2;        //X  pays  $2  as  first  horse  wins  the  rac
25  X2 =6;        //X  pays  $6  as  second  horse  wins  the  race
26  X3 =9;        //X  pays  $9  as  third  horse  wins  the  race
27  E = X1 * Ph1 + X2 * Ph2 + X3 * Ph3 ;
28  disp (E, 'expected  pay  off  for  the  race  is ')
```

Scilab code Exa 7.16 Variance and standard deviation of a random variable

```
 1  u =3;                     // mean  of  distribution  of  random
       variable  X
 2  x =[0 ,1 ,2 ,3 ,4 ,5 ,6];     // values  of  X  in  the
      distribution  as  x  where  it  is  the  number  of  times
       heads  occurs  when  a  coin  is  tossed  six  times
 3  p =[1/64 ,6/64 ,15/64 ,20/64 ,15/64 ,6/64 ,1/64];    //
```

38

```
      probabilities of occurrence of each value of X (x
      ) in the distribution such that 1/64 gives the
      probability of occurrence of no heads at all,6/64
       gives that of occurrence of heads for only one
      time and so on
4  k=0;
5  for i=1:7;
6  k=k+((x(i)-u)^2)*p(i);
7  end
8  disp(k,'Variance of X is')
9  s=sqrt(k);
10 disp(s,'Standard deviation of X is')
11
12 u=0.75;    //mean
13 x=[0,1,2,3];   //values of random variable X as x in
       the probability distribution of X
14 p=[84/220,108/220,27/220,1/220];    //probability of
       values in x which appear in distribution table of
        X
15 g=0;
16 for i=1:4;
17 g=g+((x(i))^2)*p(i);
18 end
19 h=g-(u*u);
20 disp(h,'variance of X is')
21 sd=sqrt(h);
22 disp(sd,'Standard deviation for X')
```

Scilab code Exa 7.17 Binomial diatribution

```
1  p=1/5;   //probability of the man hitting a target
2  q=1-1/5;  //probability of the man not hitting the
      target
3  n=100;   //number of times the man fires
4  e=n*p;
```

```
5  disp(e,'expected number of times the man will hit
       the target ')
6  r=sqrt(n*p*q);
7  disp(r,'Standard deviation ')
8
9  p=1/2;   //probability of guessing the correct answer
        in a five question true−false exam
10 n=5;    //number of questions in the exam
11 g=n*p;
12 disp(g,'expected number of correct answers in the
       exam ')
```

Scilab code Exa 7.18 Chebyshev inequality

```
1  u=75;     //mean of a random variable X
2  n=5;      //standard deviation of X
3  k=2;        //for k=2
4  l1=u-k*n
5  l2=u+k*n
6  P1=1-(1/k)^2
7  disp("thus the probability that a value of X lies
       between 65 and 85 is atleast 0.75 according to
       Chebyshev inequality")
8  k=3;               //for k=3
9  l1=u-k*n
10 l2=u+k*n
11 P2=1-(1/k)^2
12 disp("thus the probability that a value of X lies
       between 60 and 90 is atleast 0.8888889 according
       to Chebyshev Inequality")
```

Scilab code Exa 7.19 Sample mean and Law of large numbers

```scilab
1 disp(" a die is tossed 5 times with the following
      outcomes")
2 x1=3;
3 x2=4;
4 x3=6;
5 x4=1;
6 x5=4;
7 xmean=(x1+x2+x3+x4+x5)/5   //mean of the outcomes
8 disp('for a fair die the mean is 3.5.So law of large
       numbers tells us that as number of outcomes
      increase for this experiment,there is a greater
      likelihood that themean will get closer to 3.5')
```

Chapter 8

Graph Theory

Scilab code Exa 8.1 Paths and connectivity

```
1  // refer to page 8.6
2  disp('given a graph with 6 nodes viz. node1,node2
        ....node6')
3  A=[0 1 0 1 1 0;1 0 1 0 1 0;0 1 0 0 0 1;1 0 0 0 0 0;1
        1 0 0 0 0;0 0 1 0 0 0];
4  disp(A,'The adjacency matrix for A is')
5  disp('sequence A is a path from node4 to node6; but
        it is not a trail since the edge from node1 to
        node2 is used twice')
6  B=[0 0 0 1 1 0;0 0 0 0 1 1;0 0 0 0 0 0;1 0 0 0 0 0;1
        1 0 0 0 0;0 1 0 0 0 0];
7  disp(B,'The adjacency matrix for B is')
8  disp('sequence B is not a path since there is no
        edge from node2 to node6 is used twice')
9  C=[0 0 0 1 1 0;0 0 1 0 1 0;0 1 0 0 1 0;1 0 0 0 0 0;1
        1 1 0 0 1;0 0 0 0 1 0];
10 disp(C,'The adjacency matrix for C is')
11 disp('sequence C is a trail since is no edge is used
         twice')
12 D=[0 0 0 1 1 0;0 0 0 0 0 0;0 0 0 0 1 1;1 0 0 0 0 0;1
        0 1 0 0 0;0 0 1 0 0 0];
```

```
13 disp(D,'The adjacency matrix for D is ')
14 disp('sequence D is a simple path from node4 to
      node6 ')
```

Scilab code Exa 8.2 Minimum spanning tree

```
1  disp('to find:minimal spanning tree ')
2  disp('the adjacency matrix for the weighted graph(
      nodeA,nodeB...nodeF) of 6 nodes is :')
3  K=[0 0 7 0 4 7;0 0 8 3 7 5;7 8 0 0 6 0;0 3 0 0 0 4;4
      7 6 0 0 0;7 5 0 4 0 0]
4  disp('edges of the graph ')
5  AC=7;
6  AE=4;
7  AF=7;
8  BC=8;
9  BD=3;
10 BE=7;
11 BF=5;
12 CE=6;
13 DF=4;
14 M=[AC,AE,AF,BC,BD,BE,BF,CE,DF];  //set of all edges
15 V=int32(M);
16 L=gsort(V) //edges sorted in decreasing order of
      their weights
17 disp('deleting edges without disconnecting the graph
       until 5 edges remain ')
18 N=[BE,CE,AE,DF,BD];  //edges in minimum spanning
      tree
19 Sum=sum(N);
20 disp(Sum,'weight of the minimal spanning tree is ')
21
22
23 disp('another method of finding a minimal spanning
      tree is :')
```

```
24  K=gsort(V,'g','i')    //edges sorted in increasing
       order
25  N2=[BD,AE,DF,CE,AF];   //edges in minimum spanning
       tree
26  Sum2=sum(N2);
27  disp(Sum2,'weight of the minimal spanning tree is')
```

Chapter 9

Directed graphs

Scilab code Exa 9.6 Adjacency matrix

```
1 A =[0 0 0 1;1 0 1 1;1 0 0 1;1 0 1 0];
2 disp(A,'adjacency matrix of graph G is ')
3 A2=A^2
4 A3=A^3
5 disp('the number of ones in A is equal to the number
      of edges  in the graph i.e 8')
```

Scilab code Exa 9.8 Path matrix

```
1 A =[0 0 0 1;1 0 1 1;1 0 0 1;1 0 1 0];
2 disp(A,'adjacency matrix of graph G is ')
3 A4=A^4;
4 A3=A^3;
5 A2=A^2;
6 B4=A+A2+A3+A4;
7 B4=[4 11 7 7 0 0 0 0 3 7 4 4 4 11 7 7];
8 for i=1:16
9 if(B4(i)~=0) then
```

```
10  B4(i)=1;
11  end
12  end
13  disp(B4,'Replacing non zero entries of B4 with 1 ,we
        get path (reachability) matrix P is:')
14  disp('there are zero entries in P, therefore the
        graph is not strongly connected')
```

Chapter 11

Properties of the integers

Scilab code Exa 11.2 Division algorithm

```
1  disp('Division Algorithm')
2  a=4461;     //dividend
3  b=16;       //divisor
4  r=modulo(a,b)   //remainder
5  k=fix(a/b)      //quotient
6  j=b*k+r   //dividend=divisor*quotient+remainder
7
8  a=-262;    //dividend
9  b=3;       //divisor
10 k=fix(a/b)   //remainder
11 r=modulo(a,b)      //quotient
12 j=b*k+r   //dividend=divisor*quotient+remainder
13 disp('a and j have equal values.Hence division
       algorithm is proved')
```

Scilab code Exa 11.4 Primes

```
1  disp('Divisibility and Primes')
```

```
2 x=50;
3 disp('prime numbers less than 50 are')
4 y=primes(x)
5
6 disp('the prime factorisation of 21,24 and 1729
     respectively are:')
7 k=factor(21)
8 l=factor(24)
9 n=factor(1729)
```

Scilab code Exa 11.5 Greatest Common Divisor

```
1 disp('the GCD of the following numbers is:')
2 V=int32([12,18]);
3 [thegcd]=gcd(V)
4 V=int32([12,-18]);
5 [thegcd]=gcd(V)
6 V=int32([12,-16]);
7 [thegcd]=gcd(V)
8 V=int32([29,15]);
9 [thegcd]=gcd(V)
10 V=int32([14,49]);
11 [thegcd]=gcd(V)
```

Scilab code Exa 11.6 Euclidean algorithm

```
1 disp('Euclidean Algorithm')
2 a=[540,168,36,24];
3 b=[168,36,24,12];
4 for i=1:4
5 V=int32([a(i),b(i)]);
6 thegcd=[];
7 thegcd(i)=gcd(V);
```

```
8  disp(thegcd(i))
9  end
10
11 function []=myf(dividend,divisor)
12 quotient=floor(dividend/divisor);
13 rem=modulo(dividend,divisor);
14 k=quotient*divisor+rem;
15 disp(k)
16 if(rem~=0) then
17     myf(divisor,rem)
18 end
19 endfunction
20
21 myf(540,168)
22
23 disp('for the equation 540*x+168*y=12,we are given')
24 a=540;
25 b=168;
26 c=24;
27 d=36;
28 d=a-3*b;       //Eqn (1)
29 c=b-4*d;         //Eqn (2)
30 k=d-1*c;      //Eqn (3)
31 5*d-1*b;         //Eqn (4)
32 k=d-b+4*d;         //substituting value of c in Eqn (3)
       from Eqn (2)
33 r=5*a-16*b;
34 if(r==k) then
35     disp('x=5 and y=16');
36 end
```

Scilab code Exa 11.9 Fundamental theorem of Arithmetic

```
1  a=2^4*3^3*7*11*13
2  b=2^3*3^2*5^2*11*17
```

49

```
3  V=int32([a,b]);
4  [d]=gcd(V)
5  lcm1=2^4*3^3*5^2*7*11*13*17    //lcm is the product
       of those primes which appear in either a or b
```

Scilab code Exa 11.12 Congruence relation

```
1  x=poly(0,'x');
2  g=3*x^2-7*x+5
3  m=horner(g,2)   //value of polynomial at 2
4  n=horner(g,8)   //value of polynomial at 8
5  j=m-n
6  disp(n,"for n = ")
7  if(modulo(j,6)==0) then
8  mprintf('%i is congruent to %i(mod 6)',m,n)
9  end
```

Scilab code Exa 11.19 Linear congruence equation

```
1  disp('solving for the congruence equation 8x @ 12(
       mod 28) ,where @ is the sign for congruence')
2  a=8;
3  b=12;
4  m=28;
5  V= int32([a,m]);
6  d= gcd(V);
7  a1= a/d;
8  b1= b/d;
9  m1= m/d;
10
11 function yd= f(x)
12 yd= (a1*x)-b1
13 endfunction
```

```
14
15  disp('k is the unique solution of the equation ')
16  for i=0 : m1
17  x=i;
18  p=f(x);
19  if(modulo(p,m1) == 0)
20  k=x
21  break;
22  end
23  end
24
25  s1=k;
26  s2=k+m1;
27  s3=k+(m1*2);
28  s4=k+(m1*3);
29  disp('solutions of the original equation at d=4')
30  disp(s1)
31  disp(s2)
32  disp(s3)
33  disp(s4)
```

Chapter 12

Algebraic Systems

Scilab code Exa 12.4 Properties of operations

```
1  a=(8-4)-3
2  b=8-(4-3)
3  disp('since a and b are not equal so subtraction is
        non-commutative on Z(set of integers)')
4
5  a=[1 2;3 4]
6  b=[5 6;0 -2]
7  g= a*b
8  k= b*a
9  disp('since g and k are not equal matrix
        multiplication is non-commutative')
10
11  h=(2^2)^3
12  j=2^(2^3)
13  disp('since h and j are not equal so exponential
        operation is non associative on the set of
        positive integers N')
```

Scilab code Exa 12.17 Roots of polynomial

```
1 t=poly(0,'t');
2 f=t^3+t^2-8*t+4
3 g=factors(f)
4 disp(r=roots(f),'roots of f(t) are as follows:')
5
6 t=poly(0,'t');
7 h=t^4-2*t^3+11*t-10
8 disp(r=roots(h),'the real roots of h(t) are 1 and -2
    ')
```

Scilab code Exa 12.18 Roots of polynomial

```
1 t=poly(0,'t');
2 f=t^4-3*t^3+6*t^2+25*t-39
3 g=factors(f)
4  disp(r=roots(f),'roots of f(t) are as follows:')
```

Chapter 15

Boolean Algebra

Scilab code Exa 15.1 Basic definitions in boolean algebra

```
1  //0 denotes False and 1 denotes true
2  b=[0,1];
3  //binary operation + on the set of bits
4  for i=1:2
5  for j=1:2
6  k = b(i)& b(j);
7  disp(k)
8  end
9  end
10 //binary operation * on the set of bits
11 for i=1:2
12 for j=1:2
13 k = b(i)| b(j);
14 disp(k)
15 end
16 end
17 //unary operation ' on the set of bits
18 k=~b
19 clear;
20 D=[1,2,5,7,10,14,35,70];
21 a=35;
```

```
22  b =70;
23  V = int32 ([a,b]);
24  thelcm = lcm (V)    //a+b=lcm(a,b)
25  V = int32 ([a,b])
26  thegcd = gcd (V)    //a*b=gcd(a,b)
27  abar =70/a          //a'=70/a
```

Scilab code Exa 15.2 Boolean algebra as lattices

```
 1  D =[1 ,2 ,5 ,7 ,10 ,14 ,35 ,70];
 2  a = 2;   //a and b belong to D
 3  b = 14;
 4  V = int32 ([a,b]);
 5  thelcm = lcm (V)
 6  V = int32 ([a,b]);
 7  thegcd = gcd (V)
 8  abar =70/a
 9  bbar =70/b
10  j =[abar ,b];
11  h =[a,bbar];
12  V = int32 ([j])
13  lcm1 = lcm (V)
14  K = int32 ([h])
15  lcm2 = lcm (K)
```

Chapter 16

Recurrence relations

Scilab code Exa 16.14 Linear homogenous recurrence relations with constant coefficients

```
1  a=[];
2  a(1)=1;    //initial condition
3  a(2)=2;    //initial condition
4   disp('for recurrence relation a(n)=5*a(n-1)-4*a(n
       -2)+n^2')  //this is a second order recurrence
       relation with constant coefficients.It is non
       homogenous because of the n^2
5  for n=3:4
6  a(n)=5*a(n-1)-4*a(n-2)+n^2;
7  mprintf('Value of a(%i) is: %i \n',n,a(n))
8  end
9
10 a=[];
11 a(1)=1;    //initial condition
12 a(2)=2;    //initial condition
13 disp('for recurrence relation a(n)=2*a(n-1)*a(n-2)+n
       ^2')  //this recurrence relation is not linear
14 for n=3:4
15 a(n)=2*a(n-1)*a(n-2)+n^2;
16 mprintf('Value of a(%i) is: %i \n',n,a(n))
```

```
17 end
18
19 a=[];
20 a(1)=1;      //initial  condition
21 a(2)=2;      //initial  condition
22 disp('for  recurrence  relation  a(n)=n*a(n-1)+3*a(n-2)
      ')    //this  is  a  homogenous  linear  second  order
      recurrence  relation  without  constant  coefficients
       because  the  coefficient  of  a[n-1]  is  n,not  a
      constant
23 for  n=3:4
24 a(n)=n*a(n-1)+3*a(n-2);
25 mprintf('Value  of  a(%i)  is:  %i \n',n,a(n))
26 end
27
28
29 a=[];
30 a(1)=1;       //initial  condition
31 a(2)=2;       //initial  condition
32 a(3)=1;       //initial  condition
33 disp('for  recurrence  relation  a(n)=2*a(n-1)+5*a(n-2)
      -6*a(n-3)')  //this  is  a  homogenous  linear  third
      order  recurrence  relation  with  constant
      coefficients.Thus  we  need  three,not  two,initial
      conditions  to  yield  a  unique  solution  of  the
      recurrence  relation
34 for  n=4:6
35 a(n)=2*a(n-1)+5*a(n-2)-6*a(n-3);
36 mprintf('Value  of  a(%i)  is:  %i \n',n,a(n))
37 end
```

Scilab code Exa 16.15 Solving linear homogenous recurrence relations with constant coefficients

```
1 disp('recurrence  relation  of  Fibonacci  numbers  f[n]=
```

```
      f [ n−1]+f [ n−2] ')
 2  x=poly (0 , 'x ') ;
 3  g=x^2−x−1;
 4  disp (g , 'characterstic equation of the recurrence
        relation is : ')
 5  j =[];
 6  j=roots (g ) ;
 7  disp (j , 'roots of the characterstic equation j1 , j2 ')
 8  disp ( 'for general equation fn=Ar^n+Br^n, values of
        Aand B respectively are calculated as : ')
 9  disp ( 'initial condition at n=0 and n=1 respectively
        are : ')
10  f1=1;
11  f2=1;
12  //putting the values of f1 and f2 we get the
        equations to solve
13  D=[ 1.6180340 −0.618034;(1.6180340)^2  (−0.618034)
        ^2];
14  K=[1 1] ';
15  c =[];
16  c=D\K ;
17  A=c (1)
18  B=c (2)
19
20  disp ( 'thus the solution is f [n
        ]=0.4472136∗((1.618034)^n−(− 0.4472136)^n)] ')
```

Scilab code Exa 16.16 Solving linear homogenous recurrence relations with
constant coefficients

```
 1  disp ( 'The recurrence relation t [n]=3t [n−1]+4t [n−2] ')
 2  x=poly (0 , 'x ') ;
 3  g=x^2−3∗x−4;
 4  disp (g , 'characterstic polynomial equation for the
        above recurrence relation ')
```

```
 5  j =[];
 6  j = roots ( g );
 7  disp ( j , ' roots  of  the  characterstic  equation  j1 , j2 ')
 8  disp ( ' general  solution  t [ n]= c1 *( -1) ^ n+c2 *4^ n ) ')
 9  disp ( ' initial  condition  at  n=0  and  n=1  respectively
        are : ')
10  t0 =0;
11  t1 =5;
12  // putting  the  values  of  t0  and  t1  we  get  the
        equations  to  solve
13  D =[1  1; -1  4]
14  K =[0  5] '
15  c =[];
16  c =D \ K ;
17  c1 =c (1)
18  c2 =c (2)
19  disp ( ' thus  the  solution  is  t { n }=4^ n -( -1) ^ n ')
```

Scilab code Exa 16.17 Solving linear homogenous recurrence relations with constant coefficients

```
 1  disp ( ' The  recurrence  relation  t [ n]=4( t [ n -1] - t [ n -2]) '
        )
 2  x = poly (0 , ' x ');
 3  disp ( g =x^2 -4* x+4 , ' characterstic  polynomial  equation
        for  the  above  recurrence  relation ')
 4  j =[];
 5  j = roots ( g );
 6  disp ( j , ' roots  of  the  characterstic  equation  j1 , j2 ')
 7  disp ( ' the  general  solution  is  t [ n]= n *2^ n )
 8  disp ( ' initial  condition  at  n=0  and  n=1  respectively
        are : ')
 9  t0 =1;
10  t1 =1;
11  // putting  the  values  of  t0  and  t1  we  get  the
```

```
      equations to solve
12 D=[1 0;2 2]
13 K=[1 1] '
14 c=linsolve(D,K)
15 D=[1 0;2 2]
16 K=[1 1] '
17 c=[];
18 c=D\K;
19 c1=c(1)
20 c2=c(2)
21 disp('thus the solution is t{n}=2*n-n*2^(n-1)')
```

Scilab code Exa 16.18 Solving linear homogenous recurrence relations with constant coefficients

```
1 disp('The recurrence relation a[n]=2*a[n-1]-3a[n-2]'
     )
2 x=poly(0,'x');
3 disp(g=x^2-2*x-3,'characterstic polynomial equation
     for the above recurrence relation')
4 j=[];
5 j=roots(g);
6 disp(j,'roots of the characterstic equation j1,j2')
7 disp('the general solution is a[n]=c1*3^n+c2*(-1)^n'
     )
8 disp('initial condition at n=0 and n=1 respectively
     are:')
9 //putting the values of t0 and t1 we get the
     equations to solve
10 a0=1;
11 a1=2;
12 D=[1 1;3 -1]
13 K=[1 2] '
14 c=[];
15 c=D\K;
```

```
16   c1 = c (1)
17   c2 = c (2)
18   disp ( 'thus  the  solution  is  a [ n]=0.75*(3^ n )+0.25*(1^ n
        ) ')
```

Scilab code Exa 16.19 Solving general homogenous linear recurrence relations

```
1   disp ( 'The  recurrence  relation  a [ n]=11*a [ n−1]−39*a [ n
        −2]+45*a [ n−3] ')
2   x = poly (0 , 'x ') ;
3   disp ( g=x^3 -11*x^2+39*x -45 , 'characterstic  polynomial
        equation  for  the  above  recurrence  relation ')
4   j =[];
5   j = roots ( g ) ;
6   disp ( j , 'roots  of  the  characterstic  equation  j1 , j2 ')
7   disp ( 'hence  the  general  solution  is : a [ n]=c1 *(3^ n )+c2
        *n *(3^ n )+c3 *(5^ n ) ')
8   disp ( 'initial  condition  at  n=0  and  n=1  respectively
        are : ')
9   // putting  the  values  of  t0  and  t1  we  get  the
        equations  to  solve
10  a0 =5;
11  a1 =11;
12  a2 =25;
13  D =[1  0  1;3  3  5;9  18  25];
14  K =[5  11  25] '
15  c =[];
16  c =D \ K ;
17  c1 = c (1)
18  c2 = c (2)
19  c3 = c (3)
20  disp ( 'thus  the  solution  is  a [ n]=(4−2*n )*(3^ n )+5^n ')
```