Practical 7

Solve the following equation by thomas algorithm

$$2x_1 - x_2 = 1$$
$$-x_1 + 2x_2 - x_3 = 0$$
$$-x_2 + 2x_3 - x_4 = 0$$
$$-x_3 + 2x_4 = 1$$

Method

As we can see the equations can be written in the form of $Ax{=}r$ such as

$$A = \begin{bmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{bmatrix} \quad and \quad r = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Workout

Putting the values in R

```
rm(list=ls())
A<-matrix(c(2,-1,0,0,-1,2,-1,0,0,-1,2,-1,0,0,-1,2),nrow=4,byrow=2)
r<-c(1,0,0,1)
```

Now we will create a function *rthomas* , which takes "A" and "r" and give output the value of x , by thomas algorithm

```
rthomas<-function(A,r)
{
  n<-nrow(A)
  c<-NULL
  b<-NULL
  a<-NULL
  x<-rep(0,n)
  r<-c(1,0,0,1)
  for(i in 1:(n-1))
  {
    b<-c(b,A[i,i])            #diagonal elements
    a<-c(a,A[i+1,i])          #Lower diagonal elements
    c<-c(c,A[i,i+1])          #Upper diagonal elements
  }
  b<-c(b,A[n,n])
  a<-c(0,a)
  c<-c(c,0)
  for(i in 1:n)
  {
```

```
    if(i==1)
    {
      c[i]<-c[i]/(b[i]-a[i]*0)
      r[i]<-(r[i]-a[i]*0)/(b[i]-a[i]*0)

    }
    else
    {
      c[i]<-c[i]/(b[i]-a[i]*c[i-1])
      r[i]<-(r[i]-a[i]*r[i-1])/(b[i]-a[i]*c[i-1])
    }

  }

  for(i in n:1)
  {
    ifelse(i==n,x[i]<-r[i]-c[i]*0,x[i]<-r[i]-c[i]*x[i+1])
  }

  x

}
```

Now we will run this function on our date we will get

```
rthomas(A,r)
```

```
## [1] 1 1 1 1
```

Conclusion

1, 1, 1, 1 is solution of Ax=r