```
!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
! This code gives us the site-basis vectors for a spin-1/2 chain of size
! L with a certain fixed number of up-spins
!
! In the code: L is called "chain", the dimension is "dimTotal", and the
! site-basis vectors are "basis".
!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc

!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
!             VARIABLES to be used in the whole code
!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc

       module variables

       implicit none

       integer (kind=4) :: i,j,ii
       integer (kind=4) :: chain,upspins,dimTotal
       integer (kind=4) :: facL,facU,facLU

! SITE-BASIS
       integer (kind=4), dimension(:,:), allocatable :: basis

       end module


!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
!             Program starts here
!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc

       Program SiteBasisVectors

       use variables
       implicit none


!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
! PARAMETERS
!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc

       chain=6
       upspins=chain/2
       call SectorDimension()
       dimTotal=facL/facU
```

```
!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
!c CREATE SITE-BASIS
!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
       allocate(basis(dimTotal,chain))
       call SiteBasis()

       DO i=1,dimTotal
         write(*,*) (basis(i,j), j=1,chain)
       ENDDO

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
       deallocate(basis)


!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
!c END END END END END END END END END END END END
!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
!      STOP
       END



!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
!ccccccccccccccccc SUBROUTINES SUBROUTINES SUBROUTINES ccccccccccccc
!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc




!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
! *************** DIMENSION OF SUBSPACE ****************************
!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      subroutine SectorDimension()
      use variables
      implicit none

      facL=1
      Do i=chain-upspins+1,chain
        facL=facL*i
      Enddo

      facU=1
      Do i=2,upspins
        facU=facU*i
      Enddo
```

```
!c END of SUBROUTINE for DIMENSION OF SUBSPACE
      return
      end subroutine SectorDimension




!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
! *************** WRITING THE SITE-BASIS ***************************
!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      subroutine SiteBasis()
      use variables
      implicit none

      INTEGER (kind=4) :: ib,jb

      logical mtc
      integer (kind=4) :: in(chain),m2,h

      mtc=.false.

!c INITIALIZATION
      Do ib=1,dimTotal
        Do jb=1,chain
          basis(ib,jb)=0
        enddo
      enddo

      ii=1

71    call NEXKSB(chain,upspins,in,mtc,m2,h)
      Do jb=1,upspins
        basis(ii,in(jb))=1
      Enddo
      ii=ii+1
      if(mtc) goto 71

!c END of SUBROUTINE for SITE-BASIS
      return
      end subroutine SiteBasis
!ccccccccccccccccccccccccccccccccccccccccccccc
```

```
!c SUBROUTINE to get the COMBINATIONS
!ccccccccccccccccccccccccccccccccccccccccc
      subroutine NEXKSB(n,k,a,mtc,m2,h)
      integer (kind=4) :: n,k,a(n),m2,h,jn
      logical mtc

      if(.not.mtc) then
      m2=0
      h=k
      go to 50
      endif
      if(m2.lt.n-h) h=0
      h=h+1
      m2=a(k+1-h)
   50 do jn=1,h
      a(k+jn-h)=m2+jn
      enddo
      mtc=a(1).ne.n-k+1

      return
      end subroutine nexksb
!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
!ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
```