

```
PROGRAM part2
!*** Program to multiply together two matrices mat1 & mat2 and store
!*** the result in mat3. ie "mat3 = mat1 X mat2" where "X" represents
!*** matrix multiplication.
```

```
IMPLICIT NONE
```

```
INTEGER, PARAMETER :: m=3,n=4,k=3
REAL, DIMENSION(m,n) :: mat1
REAL, DIMENSION(n,k) :: mat2
REAL, DIMENSION(m,k) :: mat3,mat4
```

```
!**** Input the two matrices from the keyboard
!**** Press return after each row
!**** matrix one then matrix two
```

```
mat1=getmat(m,n)
mat2=getmat(n,k)
```

```
!**** Use our matrix mult. function to calc. mat3
mat3=mulmat(mat1,mat2)
```

```
!**** Perform the matrix multiplication
!**** using Fortran's MATMUL(A,B) function
```

```
mat4=MATMUL(mat1,mat2)
```

```
!**** Print out each matrix to the screen
!**** NOTE the use of / to create a blank line
```

```
PRINT '(/"Matrix One"/,)'
CALL outmat(mat1)
```

```
PRINT '(/"Matrix Two"/,)'
CALL outmat(mat2)
```

```
PRINT '(/"Matrix1 * Matrix2 (My Answer)"/,)'
CALL outmat(mat3)
```

```
PRINT '(/"Matrix1 * Matrix2 (Fortran Answer)"/,)'
CALL outmat(mat4)
```

```
CONTAINS
```

```
! *****
```

```
FUNCTION getmat(m,n)
!*** Read in matrix with "m" rows and "n" columns
```

```
INTEGER, INTENT(IN) :: m,n !**** Dummy declaration
REAL, DIMENSION(m,n) :: getmat !**** Local Declaration
```

```
INTEGER :: i ! ** Loop variable
```

```
DO i=1,m
PRINT '(("Enter matrix row :",i2)',i !*** Prompt for row number
READ*,getmat(i,:) !*** Read in row
ENDDO
```

```
END FUNCTION getmat
```

```
! *****
```

```
SUBROUTINE outmat(mat)
!*** Print any matrix out to screen
```

```
REAL, DIMENSION(:,:), INTENT(IN) :: mat !*** Dummy declaration
```

```
INTEGER :: i ! ** Loop variable
```

```
DO i=1,SIZE(mat,1) !*** Loop for each row.
PRINT*,mat(i,:) !*** Print each row.
ENDDO
```

```
END SUBROUTINE outmat
```

```
! *****
```

```
FUNCTION mulmat(mat1,mat2)
!**** Function to premultiply mat2 with mat1
```

```
!**** Use assumed shape arrays for dummy arrays****
REAL, DIMENSION(:,:), INTENT(IN) :: mat1 !*** Dummy declaration
REAL, DIMENSION(:,:), INTENT(IN) :: mat2 !*** Dummy declaration
```

```
!***** Local Declarations *****
```

```
INTEGER :: m,n,k,i,j,p
REAL, DIMENSION(SIZE(mat1,1),SIZE(mat2,2)) :: mulmat
```

```
!***** Find out the matrix sizes for the DO loop limits *****
```

```
m=SIZE(mat1,1) ; n=SIZE(mat1,2) ; k=SIZE(mat2,2)
```

```
!**** Perform the matrix multiplication
!**** using three DO loops
```

```
IF (SIZE(mat2,1) == n) THEN
DO i=1,m !*** For each row of getmat (mat3)
DO j=1,k !*** For each column of getmat (mat3)
mulmat(i,j)=0 !*** initialise to zero
DO p=1,n
mulmat(i,j)=mulmat(i,j)+mat1(i,p)*mat2(p,j)
ENDDO
ENDDO
ELSE
PRINT*,"Size mismatch in mulmat"
ENDIF
```

```
END FUNCTION mulmat
```

```
! *****
```

```
END PROGRAM part2
```

```
*****
*****
```