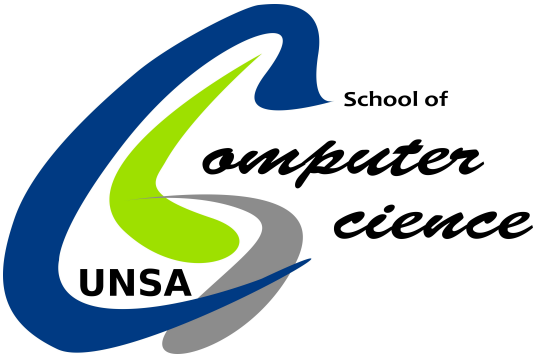
UNIVERSIDAD NACIONAL DE SAN AGUSTÍN

ESCUELA PROFESIONAL DE CIENCIA DE LA COMPUTACIÓN



TEMA:

QR ALGORITHM IN FORTRAN

Curso:

MATEMÁTICA APLICADA A LA COMPUTACIÓN

Presentado por:

Christofer Chávez Carazas

Arequipa - Perú
2017

1. Código

```
  program qr
    integer :: n,iter
    real(16), dimension(3,3) :: A,Q,R,D
    n = 3

    call zeros(Q,n)
    call zeros(R,n)

    A(1,1) = 3
    A(1,2) = 8
    A(1,3) = 1

    A(2,1) = 2
    A(2,2) = 3
    A(2,3) = 8

    A(3,1) = 0
    A(3,2) = 2
    A(3,3) = 1

    iter = 10

    D = A
    do i = 1,iter
        call qrDesc(D,n,Q,R)
        call mulMatriz(R,Q,n,D)
        call printMatriz(D,n,n)
        call zeros(Q,n)
        call zeros(R,n)
        write(*,*)

    end do

end program qr

subroutine qrDesc(A,n,Q,R)
    integer :: n,i,j
    real(16), dimension(n,n) :: A,Q,R
    real(16), dimension(n) :: aTemp,eTemp,aeTemp,↩
        uTemp
    real(16), dimension(n) :: u
    real(16) :: numTemp,normTemp

    do i = 1,n
        call getCol(A,n,i,u)
        do j = 1,i
            call getCol(Q,n,j,eTemp)
            call getCol(A,n,i,aTemp)
            call mulList(aTemp,eTemp,n,numTemp)
            R(j,i) = numTemp
            call mulListNumber(eTemp,numTemp,n,↩
                aeTemp)
            call minusList(u,aeTemp,n,uTemp)
            u = uTemp
        end do
        call norm(u,n,normTemp)
        numTemp = 1.0 / normTemp
        call mulListNumber(u,numTemp,n,eTemp)
        call copyCol(eTemp,Q,n,i)
        call getCol(Q,n,i,eTemp)
        call getCol(A,n,i,aTemp)
        call mulList(aTemp,eTemp,n,numTemp)
        R(i,i) = numTemp
    end do

return
end subroutine qrDesc

subroutine getCol(A,n,k,res)
    integer :: n,k,i
    real(16), dimension(n,n) :: A
    real(16), dimension(n) :: res

    do i = 1,n
        res(i) = A(i,k)
    end do

return
end subroutine getCol

subroutine mulList(A,B,n,res)
    integer :: n,i
    real(16) :: res
    real(16), dimension(n) :: A,B
    res = 0

    do i = 1,n
        res = res + (A(i) * B(i))
    end do

return
end subroutine mulList

subroutine mulListNumber(A,b,n,res)
    integer :: n,i
    real(16) :: b
    real(16), dimension(n) :: A,res

    do i = 1,n
        res(i) = A(i) * b
    end do

return
end subroutine mulListNumber

subroutine minusList(A,B,n,res)
    integer :: n,i
    real(16), dimension(n) :: A,B,res

    do i = 1,n
        res(i) = A(i) - B(i)
    end do

return
end subroutine minusList

subroutine norm(A,n,res)
    integer :: n,i
    real(16), dimension(n) :: A
    real(16) :: res
    res = 0

    do i = 1,n
        res = res + abs(A(i)) ** 2
    end do
    res = sqrt(res)

return
end subroutine norm

subroutine copyCol(A,B,n,k)
    integer :: n,i,k
    real(16), dimension(n,n) :: B
    real(16), dimension(n) :: A

    do i = 1,n
        B(i,k) = A(i)
    end do
return
end subroutine copyCol

subroutine mulMatriz(A,B,n,res)
    integer :: n,i,k,j
    real(16), dimension(n,n) :: A,B,res
    real(16) :: sumTemp
    do i = 1,n
        do j = 1,n
            sumTemp = 0
            do k = 1,n
                sumTemp = sumTemp + A(i,k) * B(k,j↩
                    )
            end do
            res(i,j) = sumTemp
        end do
    end do
return
end subroutine mulMatriz

subroutine printMatriz(M,f,c)
    integer :: i,j
    integer :: f,c
    real(16), dimension(f,c) :: M
    do i=1,f
        do j=1,c
            write(*,'(F10.6,$)') M(i,j)
        end do
        write(*,*)
    end do
return
end subroutine printMatriz

subroutine zeros(A,n)
    integer :: n,i,j
    real(16), dimension(n,n) :: A
    do i = 1,n
        do j = 1,n
            A(i,j) = 0
        end do
    end do
return
end subroutine zeros
```

2. Ejemplo

El ejemplo mostrado se realiza con la siguiente matriz:

$$A = \begin{bmatrix} 3 & 8 & 1 \\ 2 & 3 & 8 \\ 0 & 2 & 1 \end{bmatrix}$$

Número de iteraciones: 5

3. Resultados

Se muestra el resultado de cada iteración.

```
xnpio@xnpio-Satellite-U40t-A:~/Documentos/Xnpio/MAC/qr/fortran$ ./a.out
  7.615385  0.352090  7.202913
  1.546135 -4.150038 -0.796352
  0.000000  3.641250  3.534653

  7.519745  6.073276  3.847155
  1.096575 -1.413661  6.568801
  0.000000  0.786773  0.893916

  8.357510 -2.839235  6.012823
  0.347392 -3.992125 -4.255325
  0.000000  0.911045  2.634615

  8.232811  4.594238  4.908914
  0.165148 -2.725281  5.762904
  0.000000  0.351276  1.492470

  8.323837 -3.717351  5.525985
  0.056932 -3.488875 -5.042099
  0.000000  0.269989  2.165038

xnpio@xnpio-Satellite-U40t-A:~/Documentos/Xnpio/MAC/qr/fortran$
```