Lab Worksheet (Week-7)

Row-Reduced Echelon Form

1. Create a code in Python that can be called as a function (name it *r_r_e_f*) that gives the row reduced echelon form of any given matrix, its rank, and the pivotal columns.
   Find the rref of A and B using the above mentioned function:

$$A = \begin{bmatrix} 1 & 3 & 4 \\ 5 & -9 & -8 \\ 4 & 7 & 8 \end{bmatrix} \text{ and } B = \begin{bmatrix} 1 & 6 & 7 \\ -1 & -5 & 1 \\ 1 & 9 & 0 \end{bmatrix}$$

Algorithm

// Input: $m \times n$ matrix A

// Output:$m \times n$ matrix in reduced row echelon form, rank and the pivotal columns

1. Create a function that take the matrix A as input
2. Set i = 0, j = 0, rank = 0
3. While $i \leq m-1$ and $j \leq n-1$ (m=number of rows and n=number of columns)
4. Find position of maximum absolute value(Pivot element and position) from $A(i,j)$, $i \geq j$.
5. If pivot is equal to zero(or less than some tolerance value)
   (a) Set j = j + 1
   else
   (b) Perform $R_i \longleftrightarrow R_{pivot}$
   (c) Divide each element of row i by a_{ij}, thus making the pivot a_{ij} equal to one
   (d) For each row x from 1 to m, with $k \neq i$ subtract row i multiplied by a_{xj} from row x.
   (e) Set $i = i+1$, $j = j+1$, $rank = rank+1$;
   end if
   end while
6. Return transformed matrix A and rank.

Solution: Code for the function of rref

```
def rref(A):
    l=[]
    np.set_printoptions(suppress=True)
    m,n=np.shape(A)
    i=0;j=0;
    rank=0;
    while i≤m-1 and j≤n-1:
        p=np.max(np.abs(A[i:m,j]))
        k=np.argmax(np.abs(A[i:m,j]))
        k=k+i
        if p==0 or p≤0.000000000000001:
            j=j+1
        else:
            rank=rank+1
            C=A.copy()
            C[k,:]=A[i,:]
            C[i,:]=A[k,:]
            A=C.copy()
            A[i,:]=A[i,:]/A[i,j]
            l.append(j)
            for x in range(0,m):
                if x!=i:
                    A[x,:]=A[x,:]-A[x,j]*A[i,:]
            i=i+1
            j=j+1
    return A,rank,l
```

2. Find the basis of the Column space of the below matrices.

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & 0 & 1 \end{bmatrix} \text{ and } B = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

Solution:

```
A=np.array([[1,1,1,1],[2,1,0,1]],dtype=np.float64)
B,r,l=rref(A)
for i in l:
    print(A[:,i])
```

```
C=np.array([[1,2,3],[4,5,6],[7,8,9]],dtype=np.float64)
B,r,l=rref(C)
for i in l:
```

```
4       print(C[:,I])
```