```
    import numpy as np

    def incmatrix(genl1,genl2):
    m = len(genl1)
    n = len(genl2)
    M = None #to become the incidence matrix
    VT = np.zeros((n*m,1), int)  #dummy variable

    #compute the bitwise xor matrix
    M1 = bitxormatrix(genl1)
    M2 = np.triu(bitxormatrix(genl2),1)

    for i in range(m-1):
      for j in range(i+1, m):
        [r,c] = np.where(M2 == M1[i,j])
        for k in range(len(r)):
          VT[(i)*n + r[k]] = 1;
          VT[(i)*n + c[k]] = 1;
          VT[(j)*n + r[k]] = 1;
          VT[(j)*n + c[k]] = 1;

          if M is None:
          M = np.copy(VT)
          else:
          M = np.concatenate((M, VT), 1)

          VT = np.zeros((n*m,1), int)

    return M
```

```
    import numpy as np

    def incmatrix(genl1,genl2):
    m = len(genl1)
    n = len(genl2)
    M = None #to become the incidence matrix
    VT = np.zeros((n*m,1), int)  #dummy variable

    #compute the bitwise xor matrix
    M1 = bitxormatrix(genl1)
    M2 = np.triu(bitxormatrix(genl2),1)

    for i in range(m-1):
    for j in range(i+1, m):
    [r,c] = np.where(M2 == M1[i,j])
    for k in range(len(r)):
    VT[(i)*n + r[k]] = 1;
    VT[(i)*n + c[k]] = 1;
    VT[(j)*n + r[k]] = 1;
    VT[(j)*n + c[k]] = 1;

    if M is None:
    M = np.copy(VT)
    else:
    M = np.concatenate((M, VT), 1)

    VT = np.zeros((n*m,1), int)
```

```

    return M

```

```
    import numpy as np

    def incmatrix(genl1,genl2):
    m = len(genl1)
    n = len(genl2)
    M = None #to become the incidence matrix
    VT = np.zeros((n*m,1), int)  #dummy variable

    #compute the bitwise xor matrix
    M1 = bitxormatrix(genl1)
    M2 = np.triu(bitxormatrix(genl2),1)

    for i in range(m-1):
    for j in range(i+1, m):
    [r,c] = np.where(M2 == M1[i,j])
    for k in range(len(r)):
    VT[(i)*n + r[k]] = 1;
    VT[(i)*n + c[k]] = 1;
    VT[(j)*n + r[k]] = 1;
    VT[(j)*n + c[k]] = 1;

    if M is None:
    M = np.copy(VT)
    else:
    M = np.concatenate((M, VT), 1)

    VT = np.zeros((n*m,1), int)

    return M
```

Listing 1: Python example

```
function X = BitXorMatrix(A,B)
% function to comput the sum without charge of two vectors

    % convert elements into usigned integers
    A = uint8(A);
    B = uint8(B);

    m1 = length(A);
    m2 = length(B);
    X = uint8(zeros(m1,m2));
    for n1 = 1:m1
        for n2 = 1:m2
            X(n1,n2) = bitxor(A(n1), B(n2))
        end
    end
```