

```
function solution_array= gauss_elimination(arr,b)
%Here we have imported the matrix and the column vector
dim= size(arr);%this gives the dimension of the matrix , first column gives the number of rows
% and 2nd gives the number of rows
row_max=dim(1);
col_max=dim(2);
count=0;
%here we start our operations from 1st row to last 2nd row
for r=1:(row_max-1)
    %for having the maximum diagonal each time
    for q=r+1:row_max
        temp_col=arr(r,r);
        if(abs(arr(q,r))>abs(temp_col)) %if this comes true then we swap the rows
            %thus having the maximum value of the diagonal
            temp_row = zeros(1,col_max);
            for p=1:col_max
                temp_row(1,p)=arr(r,p);
            end
            for p=1:col_max
                arr(r,p)=arr(q,p);
            end
            for p=1:col_max
                arr(q,p)=temp_row(1,p);
            end
            temp_b=b(r,1);
            b(r,1)=b(q,1);
            b(q,1)=temp_b;
        end
    end
end

%Now since we have the maximum diagonal element in the row r, we now
%perform the gauss elimination
for y=r+1:(row_max)
    if(arr(y,r)==0)
        continue
    else
        z=arr(y,r)/arr(r,r);
        count=count+1;
    end
    for t=1:col_max
        arr(y,t)=arr(y,t)- arr(r,t)*z;
        count=count+2;
    end
    b(y,1)=b(y,1)-b(r,1)*z;
    count=count+2;
end
```

```
end

end

%now for having solution X , we convert the row echlon form matrix into row
%canonical form and then easily we can have the values
e=col_max;
while (e>1)
    i=e-1;
    while(i>0)
        z=arr(i,e)/arr(e,e);
        count=count+1;
        arr(i,e)=arr(i,e)-z*arr(e,e);
        count=count+3;
        b(i,1)=b(i,1)-z*b(e,1);
        count=count+2;
        i=i-1;
        count=count+1;
    end
    e=e-1;
end
%here we obtain the solution column vector

solution_array = zeros(row_max,1);
for i=1:col_max
    solution_array(i,1)=b(i,1)/arr(i,i);
    count=count+1;
end

return
end
```