CP.2.3.1, Sauer3

For the n-by-n matrix with entries $a_{ij} = 5/(i+2j-1)$, set $x = (1,\ldots,1)^T$ and $b = Ax$. Use the Use the python program from CP.2.2.2 or Numpy's **numpy.linalg.solve** command to compute **x_c**, the double precision computed solution. Find the infinity norm of the forward error and the error magnification factor of the problem $Ax = b$,and compare it with the condition number of A: (a) $n = 6$, (b) $n = 10$.

Hint: Be careful with Python's indexing that starts at 0, and Sauer's and Matlab's indexing that starts at 1. With zero-based indexing the formula for a_{ij} is $a_{ij} = 5/(i+2j+2)$. Here is a code snippet to generate A with $n = 5$.

```
n = 5
A = np.zeros( [ n, n ], dtype=float )
for i in range(0,n):
  for j in range(0,n):
    A[i,j] = 5 / ( i + 2*j + 2.)
print(A)
```

```
[[2.66666667 2.         1.71428571 1.55555556 1.45454545]
 [2.25       1.83333333 1.625      1.5        1.41666667]
 [2.         1.71428571 1.55555556 1.45454545 1.38461538]
 [1.83333333 1.625      1.5        1.41666667 1.35714286]
 [1.71428571 1.55555556 1.45454545 1.38461538 1.33333333]]
```