Higher Institute of Engineering & Technology, El-Beheira

Computer Engineering Department

Third assignment in numerical analysis

(The numerical solution of a system of linear equations using The Jacobi Method and The Gauss-Seidel Method.)

Under supervision of Dr.Mahmoud Gamal

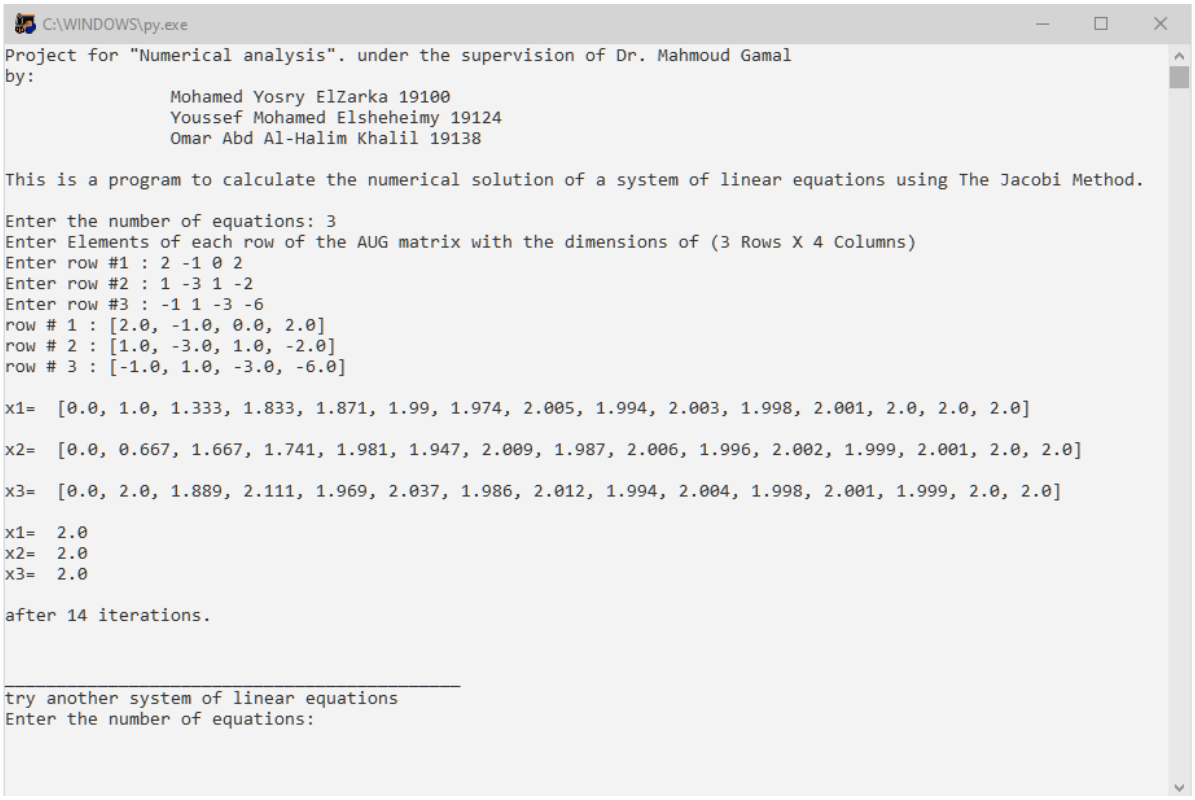
Team	**ID**
Mohamed Yosry El-Zarka	19100
Youssef Mohamed El-Sheheimy	19124
Omar Abd Al-Halim Khalil	19138

The Jacobi Method Source code in python: -

```
1 def calc_table_element(i,matrix_row,last_x):
2     res=0.0
3     res+=matrix_row[-1]
4
5     for j in range(0,len(matrix_row)-1):
6         if j==i: continue
7         res-=matrix_row[j]*last_x[j]
8
9     return res/matrix_row[i]
10
11 def check_equalization(last_x,next_x):
12     for i in range (0, len(last_x)):
13         if last_x[i] != next_x[i]:
14             return False
15     return True
16
17 print('Project for "Numerical analysis". under the supervision of
 Dr. Mahmoud Gamal')
18 print('by:')
19 print('\t\tMohamed Yosry ElZarka 19100')
20 print('\t\tYoussef Mohamed Elsheheimy 19124')
21 print('\t\tOmar Abd Al-Halim Khalil 19138\n')
22
23 print("This is a program to calculate the numerical solution of a
 system of linear equations using The Jacobi Method.\n")
24
25 while True:
26     n=int( input('Enter the number of equations: ') )
27     print("Enter Elements of each row of the AUG matrix with the
dimensions of ({} Rows X {} Columns)".format(n,n+1))
28     table=[]
29     matrix=[]
30     last_x=[]
31     next_x=[]
32     for i in range(0,n):
33         table.append([0.0])
34         last_x.append(0.0)
35         next_x.append(0.0)
36         matrix.append([]) #declare n equations
37         matrix[i]=[float(item) for item in input("Enter row #{} :
 ".format(i+1)).split()]
```

```
38          while len(matrix[i])!=n+1:
39              print("ERROR, Please enter",n+1, "items per row")
40              matrix[i].clear
41              matrix[i]=[float(item) for item in input("Enter row #
{} : ".format(i+1)).split()]
42
43      for i in range(0,n):
44          print("row #",i+1,":",matrix[i])
45
46      for iterations in range(0,200):
47          for i in range(0,n):
48              next_x[i]=calc_table_element(i, matrix[i],last_x)
49              next_x[i]=round(next_x[i],3)
50              table[i].append(next_x[i])
51          if check_equalization(last_x,next_x):
52              break
53          last_x=next_x.copy()
54
55      for i in range(0,n):
56          print ('\nx{}= '.format(i+1),table[i])
57
58      print('')
59      for i in range(0,n):
60          print('x{}= '.format(i+1),last_x[i])
61
62      print("\nafter", iterations+1 ,"iterations.")
63      print("\n\n_____________________________________________")
64      print("try another system of linear equations")
```

The Jacobi Method program



The Gauss-Seidel Method Source code in python: -

```python
def calc_table_element(i,matrix_row,recent_x):
    res=0.0
    res+=matrix_row[-1]

    for j in range(0,len(matrix_row)-1):
        if j==i: continue
        res-=matrix_row[j]*recent_x[j]

    return res/matrix_row[i]

def check_equalization(recent_x,previous_x):
    for i in range (0, len(recent_x)):
        if recent_x[i] != previous_x[i]:
            return False
    return True

print('Project for "Numerical analysis". under the supervision of Dr. Mahmoud Gamal')
print('by:')
```

```
19 print('\t\tMohamed Yosry ElZarka 19100')
20 print('\t\tYoussef Mohamed Elsheheimy 19124')
21 print('\t\tOmar Abd Al-Halim Khalil 19138\n')
22
23 print("This is a program to calculate the numerical solution of a
 system of linear equations using The Gauss-Seidel Method.\n")
24
25 while True:
26     n=int( input('Enter the number of equations: ') )
27     print("Enter Elements of each row of the AUG matrix with the
dimensions of ({} Rows X {} Columns)".format(n,n+1))
28     table=[]
29     matrix=[]
30     recent_x=[]
31     previous_x=[]
32     for i in range(0,n):
33         table.append([0.0])
34         recent_x.append(0.0)
35         previous_x.append(0.0)
36         matrix.append([]) #declare n equations
37         matrix[i]=[float(item) for item in input("Enter row #{} :
 ".format(i+1)).split()]
38         while len(matrix[i])!=n+1:
39             print("ERROR, Please enter",n+1, "items per row")
40             matrix[i].clear
41             matrix[i]=[float(item) for item in input("Enter row #
{} : ".format(i+1)).split()]
42
43     for i in range(0,n):
44         print("row #",i+1,":",matrix[i])
45
46     for iterations in range(0,200):
47         for i in range(0,n):
48             recent_x[i]=calc_table_element(i, matrix[i],recent_x)
49             recent_x[i]=round(recent_x[i],3)
50             table[i].append(recent_x[i])
51         if check_equalization(recent_x,previous_x):
52             break
53         previous_x=recent_x.copy()
54
55     for i in range(0,n):
56         print ('\nx{}= '.format(i+1),table[i])
57
```

```
58      print('')
59      for i in range(0,n):
60          print('x{}= '.format(i+1),recent_x[i])
61
62      print("\nafter", iterations+1 ,"iterations.")
63      print("\n\n_____________________________________________")
64      print("try another system of linear equations")
```

The Gauss-Seidel Method program