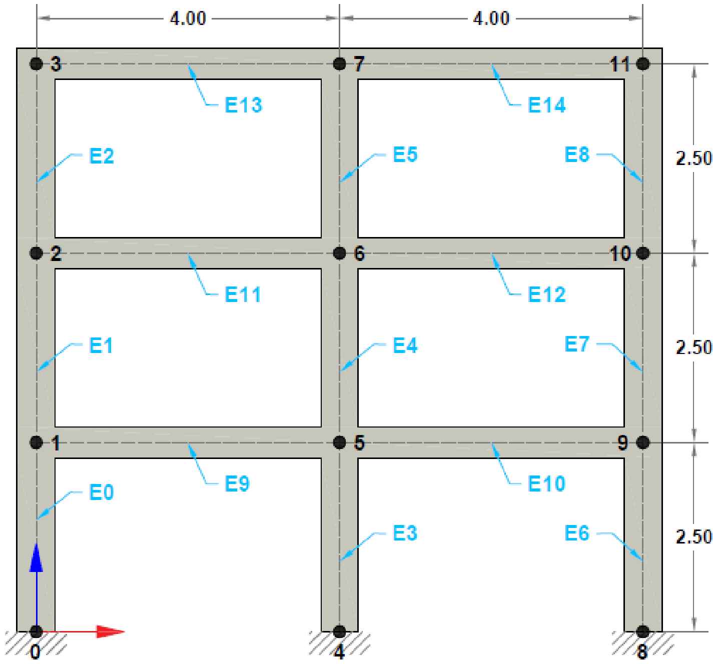
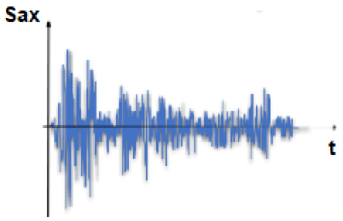
Contents under Creative Commons BY 4.0 license and code under MIT license. © Julian Parra 2019. This material is part of the Master of Engineering program by Julian Parra at Universidad EAFIT.

Linear dynamic analysis of a 2D-frame.

The two-dimensional frame shown in the figure is subjected to a base acceleration corresponding to a record from El Centro earthquake. The base acceleration is imposed along the horizontal direction. The analysis is intended to obtain the natural frequencies of the system and to find the time history of the response. The analysis conisders mass contribution only along the horizontal direction.

Input and output files for this problem are available in the examples folder of this REPO (notebooks\Examples).





- Ground acceleration signal: *El Centro earthquake record (California, 1940).*
- Time step for the excitation: *0.02 sec*
- Total time for the solution: *15.0 sec*
- Value of gravity: *9.806* m/s^2
- Element type for columns and beams: 2
- Columns cross sections *0.50 m x 0.50 m*
- Beams cross sections *0.40 m x 0.40 m*
- Material profile for all elements is concret with an elastic modulus of *2000000* $tonf/m^2$ and specific weight of *2.4* $tonf/m^3$

-Diagrams for axial and shear forces and bending moments for a selected time increment are written to the following file:
notebooks\Examples\Ex_02\Output.xls

In [1]:

```
%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np
import sympy as sym
from os import sys
sys.path.append("../source/")
from STRUCTURE import Struct_DYN
from postprocesor import *

# Execute analysis
displacement,folder,IBC,nodes,elements,ninc,T,MvarsGen,ILFGen = Struct_DYN("Examples/E
02/01_INPUT/")
```

```
 ***** Time step and seismo signal has been updated *****
 ***** Warning: Total time of seismo signal is greater than solution total time *****
----------------------
Number of nodes: 12
Number of elements: 15
Number of equations: 27
Number of equations after constraints: 27
----------------------
Natural periods of the system :  [0.14946052 0.04736539 0.02562026 0.02327047 0.020859
2 0.01848281
 0.01794195 0.01734948 0.01609275 0.01542371 0.01420094 0.01410994
 0.01373219 0.01351975 0.01271046 0.01013722 0.00932614 0.00811226
 0.00664932 0.00644066 0.0059378  0.00561157 0.00502667 0.00502665
 0.00424963 0.00363439 0.00363439]
----------------------
Time step for solution: 0.0033333333333333335 sec
Number of time increments: 6000
----------------------
Finished initial conditions....: 0
Duration for system solution: 0:00:13.663503
Duration for the system's solution: 0:00:13.665530
Duration for post processing: 0:00:00
----------------------
Analysis terminated successfully!
----------------------
```

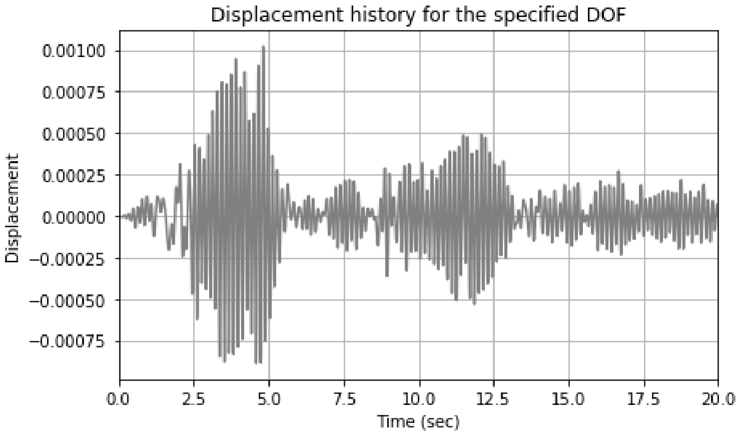
Results

Notice that in this analysis the time step initially specified for the analysis $\Delta t = 0.02s$ is too large for convergence thus forcing nldyna to adjust the time step.

The horizontal displacement time history at nodal point 3 is shown below:

In [2]:

```
fig = NodalDispPLT(displacement[6 ,:], T, ninc, ylabel = "Displacement")
```



In [3]:

```
from IPython.core.display import HTML
def css_styling():
    styles = open('./nb_style.css', 'r').read()
    return HTML(styles)
css_styling()
```

Out[3]:

In [ ]: