Available online www.ejaet.com

European Journal of Advances in Engineering and Technology, 2024, 11(9):23-43

Research Article ISSN: 2394 - 658X

WAFEC- An Innovative Web-based Computer Application for Finite Element Analysis Using Direct Stiffness Method in JAVA™/J2EE™

Arfath Mohammad[1], Rajeev Nair[2], Hamid Lankarani[3]

[1,2,3]Mechanical Engineering Department, Wichita State University, Wichita, Kansas, USA-67226
[2]rajeev.nair@wichita.edu

ABSTRACT

This study aims at demonstrating the development and implementation of a web-based computer software, WAFEC (Web Application for Finite Element Calculations) created using JAVA, J2EE programming environment to solve finite element problems related to structural engineering by utilizing direct stiffness method approach. It is designed to perform modeling of the global stiffness matrix, and calculate deformation and forces for one-, two- and three-dimensional finite elements by overcoming limitations of existing software's. Most of the existing finite element software's are desktop applications and they require prior installation, basic understanding of the software before use, and are time consuming which makes it difficult to get an instant solution for the user.

Obtaining stiffness matrix using this approach, is not only difficult but also not easily comprehensible, because the results obtained are not in a proper readable format. More ever most of the software's are commercial software's hat are more oriented towards industrial applications rather than academic usage. The major purpose of this paper is to develop a computer software to solve problems with infinite nodes and elements in a limited amount of time and to increase the flexibility of use with the help of internet. The robust code is designed to perform analytical calculations for elements in different complex connection structures. The architecture of software created is based on MVC framework which is formed by utilizing Java Server Pages to form user interface that helps to perform data entry, Servlet to perform stiffness calculation and MySQL to store the results. Application of proposed web-based software in solving practical problems is demonstrated by carrying on several case studies utilizing different finite elements and comparing it to a few other software's based on student feedback.

Keywords: Finite Element Method, Global Stiffness Matrix, Deformation, Forces, Java/J2EE, Web Application

Nomenclature:

K	=	Stiffness of Spring
K	=	Global Stiffness Matrix
A	=	Area of Element
E	=	Modulus of Elasticity
I	=	Principal Moment of Inertia
L	=	Length of Element
J	=	Jacobian Matrix

INTRODUCTION

Since past several decades, the problems related to structural and non-structural aspects of engineering are solved using the analytical approach. But due to complexities in geometry of the structure, it becomes impossible to analyze the geometry using this approach. In order to overcome this limitation numerical method was introduced. Several approaches are utilized to solve problems by numerical approach method. Out of those Finite Element Method is considered as one of the most effective methods. This method involves the process of discretizing the geometry into smaller units called finite elements that makes it easy to obtain solution for the complex cases.

Finite element analysis problems that requires complicated matrix multiplication can be solved instantly by creating a computer program. But creating a program is very complex and difficult as it requires diverse background knowledge including mechanics, mathematics and computer science. Currently a variety of computer software's both commercial and non-commercial, exists to help perform finite element analysis. Most of the software's are developed using procedural languages. The procedural approach has been proven effective in treating algorithmic complexity. However, such an approach does not address design and quality issues of the overall program [1]. In addition to that there are a lot of disadvantages associated with this software's. Some of them are as follows:

1) The cost of license for commercial FEA packages are usually high
2) Most of the FEA packages installed requires regular maintenance and software upgrades. Any upgrade setting for the FEA package could conflict with any other computer settings caused by other existing software. In addition to it these software upgrades forces a computer restart in order to bring new changes into effect.
3) It's time consuming to feed information to the software in order to define the finite element model.
4) Most of the commercial software's have restriction to access the core functionality of the program. Thus, understanding the internal program code becomes difficult which causes great difficulty to extend or customize the software.
5) Depending on the geometry the time required for execution of the defined model in terms of CPU to generate results is usually high.
6) Most of the commercial software's are desktop applications which require prior installation and restrict using it on other computers other than the one which it is installed.
7) A desired platform, operating system specific configuration and computer memory space are pre-requisites to install desktop FEA packages.
8) Most of the FEA packages are complex and requires proper training prior to its usage.
9) Most of the FEA packages are general purpose software's due to which their approach to solve different types of categories of problems requires same type of steps and validations. But most of the academic problems does not require most of those complex validations or complex modeling in order to get simple solution.
10) Certain FEA packages are restricted to solve problems by discretizing the model into a restricted number of elements.
11) Providing wrong input while defining a model will lead to repeating the tedious process to define model again.
12) Most of the commercial purpose software's are designed for industrial use, and so their graphical interface is very complex, which makes them academic user "unfriendly".
13) Most of the software's are created using procedural languages which results in less efficiency and poor performance of application while impacting computer performance.
14) Readability of the solution generated by most of the existing software's is complex and lack instant understanding.
15) Most of the FEA packages requires geometric modelling for each problem.

So, there is a strong need for a special purpose finite element software that can overcome the above limitations. This was the motivation for the creation of web based tool for finite element method calculations called WAFEC. The major objective here is to create a special purpose web software that can be used to solve problems related to one, two and three dimensional finite elements consisting of any element numbers in order to help students understand concepts of finite element analysis and solve problems related to structural engineering with emphasis on methods rather than focus on cost, installation, geometric modelling, system requirements, complex user interface or CPU time.

Literature Review

The initial development of finite element was started in 1941 by Hrennikoff [21]. Later following the advancement of research conducted by several authors several resources were published. Direct stiffness method was introduced by Turner et al. [22] in 1956 and the first-time finite element keyword was used by Clough [23] in 1960. In order to solve problems related to finite element several programming languages are utilized to develop finite element packages and programs. Initially most of the finite element programs were developed using FORTRAN. But since FORTRAN is a procedural language, it becomes very tough in code extension, maintenance and reusability. This leads to introduction of object-oriented programming in finite element. Several packages were developed for commercial and non-commercial use in solving finite element problems. Academically MATLAB [24] programming is used widely to solve finite element problems. Several research papers were published describing the implementation of finite element modeling using object-oriented programming and MATLAB [1-5]. Several other commercial softwares like ANSYS [25], Algor [26], MSC/NASTRAN [27], STARDYNE [28] are widely used in solving large scale finite element problems. In addition to commercial there are many open source softwares like Maxima, FreeMat, LISA which helps in performing finite element analysis.

Several books are published to explain about finite element methods [5-10] and several research investigations are done [11-15]. One such book is “A First Course in the Finite Element Method” [20] which discusses the finite element modelling and analysis of one-, two- and three-dimensional elements. The book is very important in understanding the concepts related to to direct stiffness method in detail. It has separate chapters for different type of elements as well as various level of problems for each type of element.

COMPUTATIONAL EXPERIMENT

Methodology

The aim of this research is to create a web application software in Java/J2EE Technologies. The purpose of creating a web application is to make it accessible at all locations without prior installation. It is created by keeping in mind the three basic steps of finite element analysis shown in figure below.

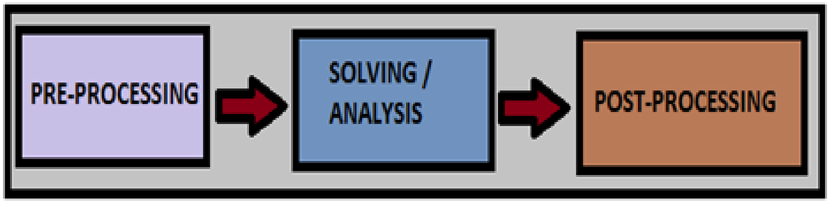


***Figure 1** - Finite Element Analysis Steps*

Pre-Processing: During this stage the complex geometry is broken into finite elements and input information regarding each element is fed into the software. The different types of input used is discussed in detail in section 2.1 below.

Solving/Analysis: The data from the preprocessing stage is used in this stage and is then converted into algebraic equations by writing it in the form of an equation as shown below.

$$F = Kd$$

The equation is then solved and unknowns are determined.

Post-Processing: During this stage results from the Solving/analysis stage is available for the user.

Input

The input required to define the structure varies from element to element. The Figure 1 describes about various levels of inputs that will be used to describe the finite element structure.

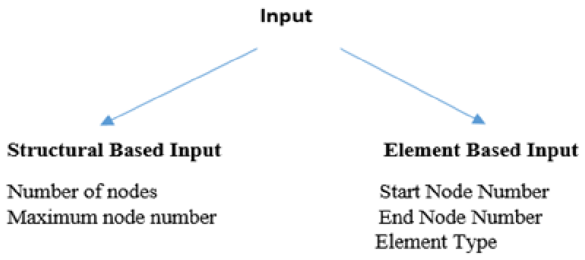


***Figure 2** - Input Model*

Structural based input parameters remain same when describing the structure in the web application but the information required to describe each element of the structure varies from element to element. In order to describe an element apart from start node and end number other the parameters are required which are listed in figure.

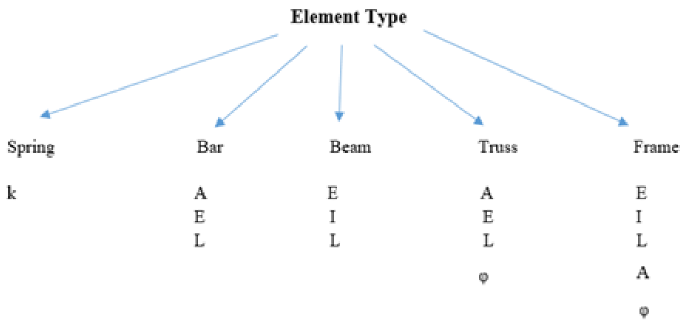
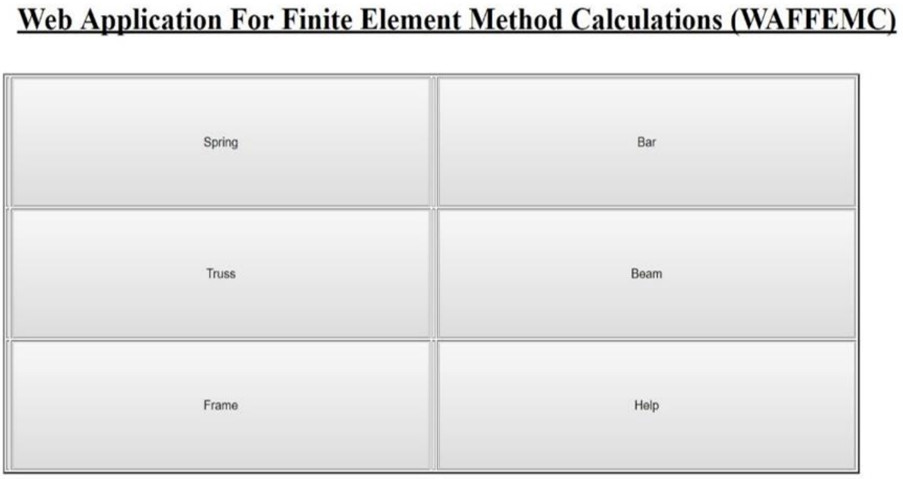


***Figure 3** - Input Element Classification*

The input to the web application is provided with the help of JSP pages. The different types of input web pages for the web application is described in user interface section.

User Interface

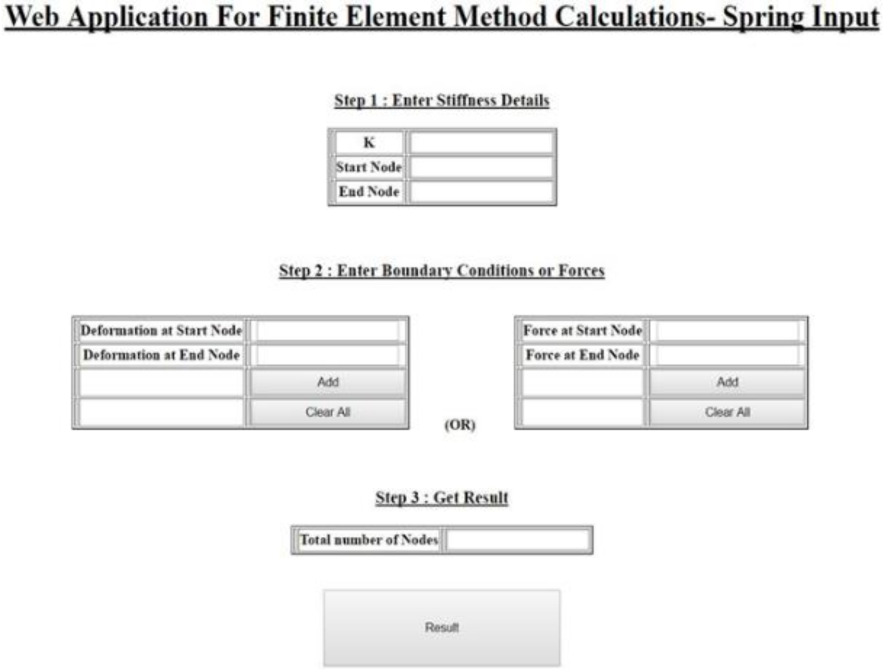


Web Application For Finite Element Method Calculations (WAFFEMC)

Figure 4 - *Web Application Launch Screen*

When the Web application is launched, it will take users to home screen. Home screen consist of various options for various elements. On clicking of a specific element, it will redirect you to JSP page for that specific element. This redirection to different types of pages based on clicking the specific element name is done with the help of a servlet. Separate Help section is provided to help users in navigation as well as to provide a quick tutorial about the basic concepts of finite element analysis.

On clicking spring button from index page, it will redirect to spring web page as shown in figure. This page provides information to take stiffness details and start node from where element starts and end node to where the element ends. In addition to it, it asks for boundary conditions at that node or any force mentioned for that node. After filling details when user click Add button it will redirect to SpringServlet. SpringServlet is a servlet which takes input from jsp page and performs the stiffness operation, as well as force and deformation.



Web Application For Finite Element Method Calculations- Spring Input

Step 1 : Enter Stiffness Details

K	
Start Node	
End Node	

Step 2 : Enter Boundary Conditions or Forces

Deformation at Start Node	
Deformation at End Node	

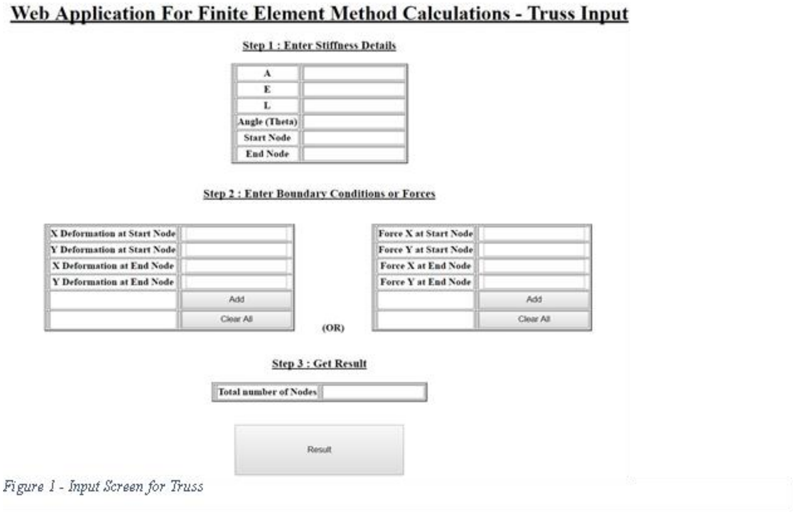
(OR)

Force at Start Node	
Force at End Node	

Step 3 : Get Result

Total number of Nodes	

Figure 5 - *Input Screen for spring*



Web Application For Finite Element Method Calculations - Truss Input

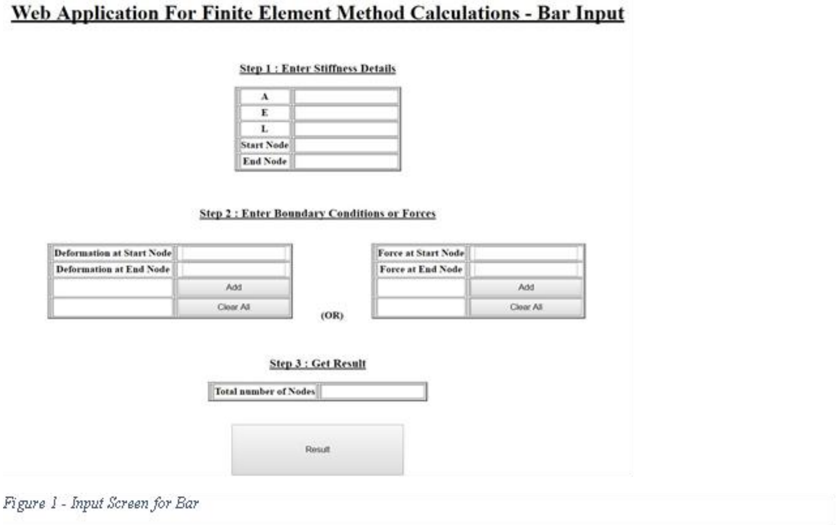
Step 1 : Enter Stiffness Details

Step 2 : Enter Boundary Conditions or Forces

Step 3 : Get Result

Figure 1 - Input Screen for Truss

Figure 6 *- Input Screen for Truss*



Web Application For Finite Element Method Calculations - Bar Input

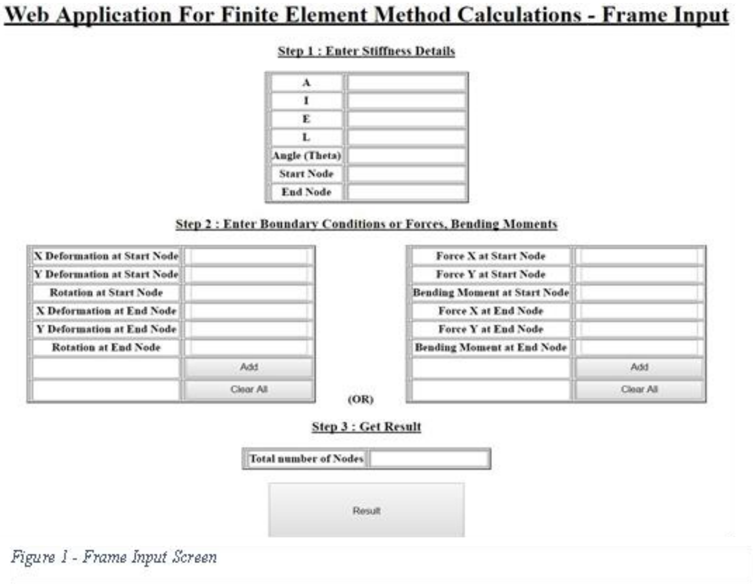
Step 1 : Enter Stiffness Details

Step 2 : Enter Boundary Conditions or Forces

Step 3 : Get Result

Figure 1 - Input Screen for Bar

Figure 7 *- Input Screen for Bar*



Web Application For Finite Element Method Calculations - Frame Input

Step 1 : Enter Stiffness Details

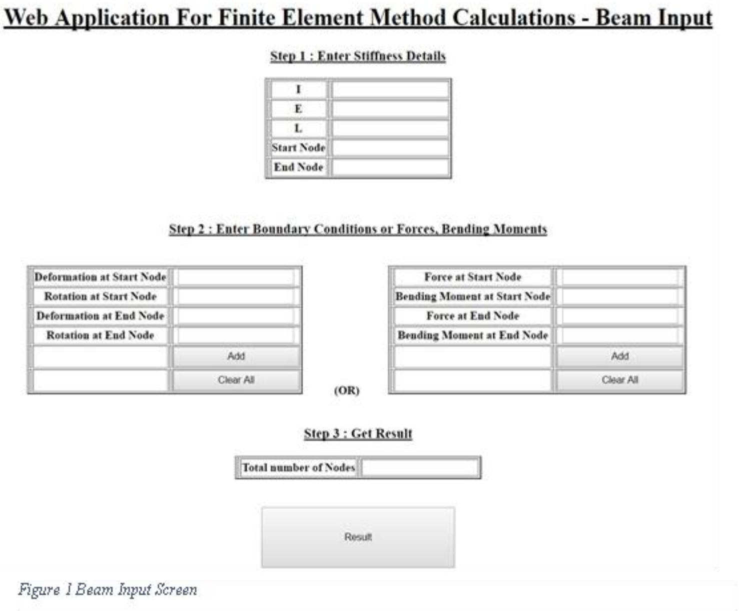
Step 2 : Enter Boundary Conditions or Forces, Bending Moments

X Deformation at Start Node
Y Deformation at Start Node
Rotation at Start Node
X Deformation at End Node
Y Deformation at End Node
Rotation at End Node
Add
Clear All
(OR)
Force X at Start Node
Force Y at Start Node
Bending Moment at Start Node
Force X at End Node
Force Y at End Node
Bending Moment at End Node
Add
Clear All

Step 3 : Get Result

Figure 1 - Frame Input Screen

Figure 8 *- Frame Input Screen*



Web Application For Finite Element Method Calculations - Beam Input

Step 1 : Enter Stiffness Details

I	
E	
L	
Start Node	
End Node	

Step 2 : Enter Boundary Conditions or Forces, Bending Moments

Deformation at Start Node	
Rotation at Start Node	
Deformation at End Node	
Rotation at End Node	

(OR)

Force at Start Node	
Bending Moment at Start Node	
Force at End Node	
Bending Moment at End Node	

Step 3 : Get Result

Total number of Nodes	

Figure 1 Beam Input Screen

Figure 9 *Beam Input Screen*

Object Oriented Programming:

In order to overcome the limitations of procedural programming. Object Oriented Programming is introduced. In object oriented programming everything revolves around object. Object consist of data type and method which defines the state and behavior of object respectively. Object are created for classes. Class is a prototype or template from which object is created.

Pillars of Object Oriented Programming:

1) Abstraction
2) Inheritance
3) Encapsulation
4) Polymorphism

Advantages of using Object Oriented Programming in developing our software:

1) The code for each object can be developed and maintained separately which helps in easy extending of code and the features of software in future, which is a limitation in some of the existing software.

2) It helps in data hiding which helps in increasing the security of the system as interaction happens only to object methods.

3) It helps in code reusability in order to implement any enhancement in future.

4) It helps in easy design of the complex programs compared to procedural programming.

5) Very less maintenance is required to maintain the program.

Our software is developed based on the principles of object oriented programming. All the concepts related to abstraction, inheritance, encapsulation and polymorphism is applied to increase code reusability, maintainability, security and to reduce complexity. SpringServlet, BarServlet, BeamServlet, TrussServlet, FrameServlet complexity is reduced by using Object oriented programming approach in our software.

Program Structure

The program structure is based on Model View Control architecture which consist of three major components as shown in below figure.

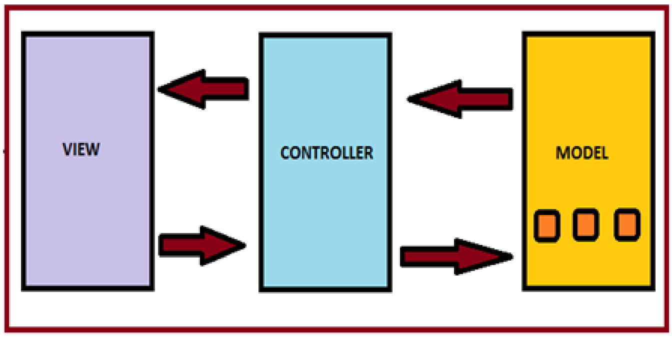


Figure 10 *- Architecture of Software*

View: As soon as user enters the web application URL it will redirect him to the view component of web application. View component is basically presentation component which helps to present the data to user. All the user interface pages are part of view component. They are designed using JSP technology of J2EE. The various JSP pages and their function is listed below.

Table 1 - JSP Pages Information

S.no	Name	Description
1	Index.jsp	This is the home web page of the web application
2	SpringInput.jsp	This web page is used to take spring input from users
3	SpringResults.jsp	This web page is used to display results of spring calculations
4	BarInput.jsp	This web page is used to take bar input from users
5	BarResults.jsp	This web page is used to display results of bar calculations
6	BeamInput.jsp	This web page is used to take beam input from users
7	BeamResults.jsp	This web page is used to display results of beam calculations
8	TrussInput.jsp	This web page is used to take truss input from users
9	TrussResults.jsp	This web page is used to display results of truss calculations
10	FrameInput.jsp	This web page is used to take frame input from users
11	FrameResults.jsp	This web page is used to display results of frame calculations
12	Help.jsp	This screen provides information about formulae's and step by step tutorials to guide users

Controller: Controllers are invoked based on the user selection and user input from the jsp pages. It is the middle layer between view layer and model layer and it performs interactions between them. Based on the input from view layer it stores the input in model layer and then process the data by perform the respective operation to calculate stiffness, deformation and forces and send data back to view layer based on user selection. Servlet technology of J2EE is used to design the controller layer of the web application. The various servlets and their functions are listed in below table:

Table 2 - Servlets Information

S.no	Name	Description
1	RedirectServlet	This servlet is used to redirect the user to respective element jsp page based on the selection
2	SpringServlet	This servlet is used to take input from SpringInput.jsp perform operations
3	BarServlet	This web page is used to display results of spring calculations
4	BeamServlet	This web page is used to take bar input from users
5	TrussServlet	This web page is used to display results of bar calculations
6	FrameServlet	This web page is used to take beam input from users

Model: This is the persistence layer of the web application. It serves requests of other two layers and store the information. It is designed using MySQL database. It consists of 15 tables. 5 tables to calculate stiffness of elements, 5 tables to perform deformation calculation and remaining 5 tables helps in calculation of forces. Columns of each table depends on the input described in the input model.

The interaction of various layers in the web application is shown in the below figure:

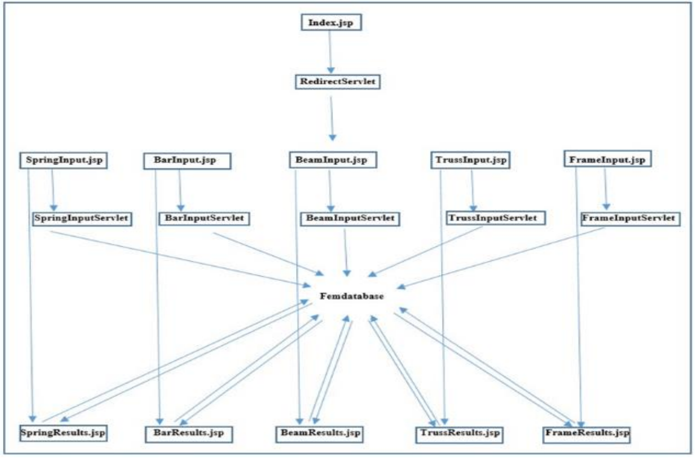


***Figure 11 -** Flow Chart of Application*

THEORY & TERMINOLOGY

Finite Element Method

Numerical analysis of structural problems based on Finite Element Methods (FEM) requires some basic knowledge of matrix operations [2]. The Finite Element Method (FEM) is one of the most powerful tools used in structural analysis [3]. FEM is a method for dividing up a very complicated problem into small elements that can be solved in relation to each other. [4] These small elements are called finite elements. Each finite element is joined to other finite element by using nodes or by using a common surface resulting in the formation of the structure. It is basically a numerical approach for solving engineering problems. In this approach, each finite element of the structure gives an algebraic equation and combining all the algebraic equations and solving gives the solution of the structure. The result obtained is an approximate result. Finite element analysis is most commonly done by direct stiffness method. Several books are published to explain about finite element methods [5-10] and several research investigations are done [11-15].

Direct Stiffness Method

It is based on the concept that the actual structure can be idealized as a set of finite elements connecting nodal points. [16] It is a step by step approach for solving finite element method problems. In this process stiffness matrix of individual elements is superimposed to generate global stiffness matrix. This is then solved by applying boundary conditions in order to get displacement and other unknowns.

Nodes:

These are defined as the points that indicates starting point and end point of an element. Number of nodes does not remain constant and varies from one element type to another.

Series Connection:

Two elements are termed as to be in series connection if one end of both the element share a common node.

Below Figure shows two spring elements in series. Element 1 and Element 2 one end is sharing node2

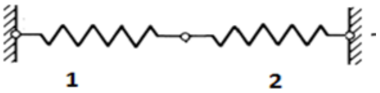


Figure 12 *- Series Connection*

Parallel Connection:

Two elements are termed to be in parallel connection if both ends of the elements are parallel to each other as shown in below figure.

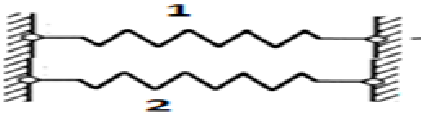


Figure 13 *- Parallel Connection*

Boundary Conditions

A model developed is usually free model untilconstrains are applied on it to restrict its movement. These restrictions are termed as boundary conditions.

In the finite element method boundary conditions are used to either form force vectors (natural or Neumann boundary conditions) or to specify the value of the unknown field on a boundary (essential or Dirichlet boundary conditions) [17]. They are used on the global stiffness matrix in order to solve the algebraic equations more easily. Some of the most commonly used boundary conditions in problems related to structural engineering are:

Table 3 - Boundary Conditions

S.no	Type of Boundary Condition	Description
1	Fixed Support	Displacement in all directions and rotations at fixed support node is zero
2	Pin Support	Displacement in the particular axis will be zero
3	Roller Support	Displacement in the particular axis will be zero

Spring:

Spring is considered as one of the basic type of finite element. Stiffness matrix for spring is given by

$$k = \begin{bmatrix} k & -k \\ -k & k \end{bmatrix}$$

Global Stiffness Matrix for spring elements in series:

Consider stiffness matrix for element 1 as

$$k1 = \begin{bmatrix} k1 & -k1 \\ -k1 & k1 \end{bmatrix}$$

Consider stiffness matrix for element 2 as

$$k2 = \begin{bmatrix} k2 & -k2 \\ -k2 & k2 \end{bmatrix}$$

Then Global Stiffness Matrix:

$$K = \begin{bmatrix} k1 & -k1 & 0 \\ -k1 & k1+k2 & -k2 \\ 0 & -k2 & k2 \end{bmatrix}$$

Global Stiffness Matrix for spring element in parallel:

$$K = \begin{bmatrix} k1+k2 & -k1-k2 \\ -k1-k2 & k1+k2 \end{bmatrix}$$

Bar:

Stiffness matrix for bar element is given by:

$$k = AE/L \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

Consider stiffness matrix for element 1 as

$$k_1 = A_1E_1/L_1 \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

Consider stiffness matrix for element 2

$$k_2 = A_2E_2/L_2 \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

Then Global Stiffness Matrix for bar elements in series:

$$K = \begin{bmatrix} (A_1E_1/L_1) & -(A_1E_1/L_1) & 0 \\ -(A_1E_1/L_1) & (A_1E_1/L_1)+(A_2E_2/L_2) & -(A_2E_2/L_2) \\ 0 & -(A_2E_2/L_2) & (A_2E_2/L_2) \end{bmatrix}$$

Global Stiffness Matrix for bar element in parallel:

$$K = \begin{bmatrix} (A_1E_1/L_1)+(A_2E_2/L_2) & -((A_1E_1/L_1)+(A_2E_2/L_2)) \\ -((A_1E_1/L_1)+(A_2E_2/L_2)) & (A_1E_1/L_1)+(A_2E_2/L_2) \end{bmatrix}$$

Beam:

Beams and beam like elements are main constituent of structures and widely used in aerospace, high speed machinery, light weight structure, etc. and experience a wide variety of static and dynamic loads of certain frequency of vibration which leads to its failure due to resonance [18]. Stiffness matrix for beam is given by:

$$k = \frac{EI}{L}\begin{bmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12 & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{bmatrix}$$

Global Stiffness Matrix for beam elements in series:

Consider stiffness matrix for element 1 as

$$k_1 = \frac{E_1I_1}{{L_1}^3}\begin{bmatrix} 12 & 6L_1 & -12 & 6L_1 \\ 6L_1 & 4{L_1}^2 & -6L_1 & 2{L_1}^2 \\ -12 & -6L_1 & 12 & -6L_1 \\ 6L_1 & 2{L_1}^2 & -6L_1 & 4{L_1}^2 \end{bmatrix}$$

Consider stiffness matrix for element 2 as

$$k_2 = \frac{E_2I_2}{{L_2}^3}\begin{bmatrix} 12 & 6L_2 & -12 & 6L_2 \\ 6L_2 & 4{L_2}^2 & -6L_2 & 2{L_2}^2 \\ -12 & -6L_2 & 12 & -6L_2 \\ 6L_2 & 2{L_2}^2 & -6L_2 & 4{L_2}^2 \end{bmatrix}$$

Then Global Stiffness Matrix: K =

$$\begin{bmatrix} 12\frac{E_1I_1}{{L_1}^3} & 6\frac{E_1I_1}{{L_1}^2} & -12\frac{E_1I_1}{{L_1}^3} & 6\frac{E_1I_1}{{L_1}^2} & 0 & 0 \\ 6\frac{E_1I_1}{{L_1}^2} & 4\frac{E_1I_1}{L_1} & -6\frac{E_1I_1}{{L_1}^2} & 2\frac{E_1I_1}{L_1} & 0 & 0 \\ -12\frac{E_1I_1}{{L_1}^3} & -6\frac{E_1I_1}{{L_1}^2} & 12\frac{E_1I_1}{{L_1}^3}+12\frac{E_2I_2}{{L_2}^3} & -6\frac{E_1I_1}{{L_1}^2}+6\frac{E_2I_2}{{L_2}^2} & -12\frac{E_2I_2}{{L_2}^3} & 6\frac{E_2I_2}{{L_2}^2} \\ 6\frac{E_1I_1}{{L_1}^2} & 2\frac{E_1I_1}{L_1} & -6\frac{E_1I_1}{{L_1}^2}+6\frac{E_2I_2}{{L_2}^2} & 4\frac{E_1I_1}{L_1}+4\frac{E_2I_2}{L_2} & -6\frac{E_2I_2}{{L_2}^2} & 2\frac{E_2I_2}{L_2} \\ 0 & 0 & -12\frac{E_2I_2}{{L_2}^3} & -6\frac{E_2I_2}{{L_2}^2} & 12\frac{E_2I_2}{{L_2}^3} & -6\frac{E_2I_2}{{L_2}^2} \\ 0 & 0 & 6\frac{E_2I_2}{{L_2}^2} & 2\frac{E_2I_2}{L_2} & -6\frac{E_2I_2}{{L_2}^2} & 4\frac{E_2I_2}{L_2} \end{bmatrix}$$

Global Stiffness Matrix for beam element in parallel:
K =

$$\begin{bmatrix} 12\frac{E_1I_1}{{L_1}^3}+12\frac{E_2I_2}{{L_2}^3} & 6\frac{E_1I_1}{{L_1}^2}+6\frac{E_2I_2}{{L_2}^2} & -(12\frac{E_1I_1}{{L_1}^3}+12\frac{E_2I_2}{{L_2}^3}) & 6\frac{E_1I_1}{{L_1}^2}+6\frac{E_2I_2}{{L_2}^2} \\ 6\frac{E_1I_1}{{L_1}^2}+6\frac{E_2I_2}{{L_2}^2} & 4\frac{E_1I_1}{L_1}+4\frac{E_2I_2}{L_2} & -(6\frac{E_1I_1}{{L_1}^2}+6\frac{E_2I_2}{{L_2}^2}) & 2\frac{E_1I_1}{L_1}+2\frac{E_2I_2}{L_2} \\ -(12\frac{E_1I_1}{{L_1}^3}+12\frac{E_2I_2}{{L_2}^3}) & -(6\frac{E_1I_1}{{L_1}^2}+6\frac{E_2I_2}{{L_2}^2}) & 12\frac{E_1I_1}{{L_1}^3}+12\frac{E_2I_2}{{L_2}^3} & -(6\frac{E_1I_1}{{L_1}^2}+6\frac{E_2I_2}{{L_2}^2}) \\ 6\frac{E_1I_1}{{L_1}^2}+6\frac{E_2I_2}{{L_2}^2} & 2\frac{E_1I_1}{L_1}+2\frac{E_2I_2}{L_2} & -(6\frac{E_1I_1}{{L_1}^2}+6\frac{E_2I_2}{{L_2}^2}) & 4\frac{E_1I_1}{L_1}+4\frac{E_2I_2}{L_2} \end{bmatrix}$$

Truss:

Generally, the bar elements are applied to construct the truss structures [19].

Stiffness matrix for truss is given by k =

$$\frac{AE}{L}\begin{bmatrix} \cos\theta^2 & \cos\theta\sin\theta & -\cos\theta^2 & -\cos\theta\sin\theta \\ \cos\theta\sin\theta & \sin\theta^2 & -\cos\theta\sin\theta & -\sin\theta^2 \\ -\cos\theta^2 & -\cos\theta\sin\theta & \cos\theta^2 & -\cos\theta\sin\theta \\ -\cos\theta\sin\theta & -\sin\theta^2 & -\cos\theta\sin\theta & \sin\theta^2 \end{bmatrix}$$

Global Stiffness Matrix for spring elements in series:

Let $\cos\theta_1 = C_1$, $\cos\theta_2 = C_2$, $\sin\theta_1 = S_1$, $\sin\theta_1 = S_2$

Consider stiffness matrix for element 1 as

$$k_1 = \begin{bmatrix} \frac{A_1E_1}{L_1}{C_1}^2 & \frac{A_1E_1}{L_1}C_1S_1 & -\frac{A_1E_1}{L_1}{C_1}^2 & -\frac{A_1E_1}{L_1}C_1S_1 \\ \frac{A_1E_1}{L_1}C_1S_1 & \frac{A_1E_1}{L_1}{S_1}^2 & -\frac{A_1E_1}{L_1}C_1S_1 & -\frac{A_1E_1}{L_1}{S_1}^2 \\ -\frac{A_1E_1}{L_1}{C_1}^2 & -\frac{A_1E_1}{L_1}C_1S_1 & \frac{A_1E_1}{L_1}{C_1}^2 & -\frac{A_1E_1}{L_1}C_1S_1 \\ -\frac{A_1E_1}{L_1}C_1S_1 & -\frac{A_1E_1}{L_1}{S_1}^2 & -\frac{A_1E_1}{L_1}C_1S_1 & \frac{A_1E_1}{L_1}{S_1}^2 \end{bmatrix}$$

Consider stiffness matrix for element 2 as

k_2 =

$$\begin{bmatrix} \frac{A_2E_2}{L_2}{C_2}^2 & \frac{A_2E_2}{L_2}C_2S_2 & -\frac{A_2E_2}{L_2}{C_2}^2 & -\frac{A_2E_2}{L_2}C_2S_2 \\ \frac{A_2E_2}{L_2}C_2S_2 & \frac{A_2E_2}{L_2}{S_2}^2 & -\frac{A_2E_2}{L_2}C_2S_2 & -\frac{A_2E_2}{L_2}{S_2}^2 \\ -\frac{A_2E_2}{L_2}{C_2}^2 & -\frac{A_2E_2}{L_2}C_2S_2 & \frac{A_2E_2}{L_2}{C_2}^2 & -\frac{A_2E_2}{L_2}C_2S_2 \\ -\frac{A_2E_2}{L_2}C_2S_2 & -\frac{A_2E_2}{L_2}{S_2}^2 & -\frac{A_2E_2}{L_2}C_2S_2 & \frac{A_2E_2}{L_2}{S_2}^2 \end{bmatrix}$$

Then Global Stiffness Matrix for truss elements in series:

K =

$$\left[\begin{matrix} \frac{A_1E_1}{L_1}C_1^2 & \frac{A_1E_1}{L_1}C_1S_1 & -\frac{A_1E_1}{L_1}C_1^2 \\ \frac{A_1E_1}{L_1}C_1S_1 & \frac{A_1E_1}{L_1}S_1^2 & -\frac{A_1E_1}{L_1}C_1S_1 \\ -\frac{A_1E_1}{L_1}C_1^2 & -\frac{A_1E_1}{L_1}C_1S_1 & \frac{A_1E_1}{L_1}C_1^2+\frac{A_2E_2}{L_2}C_2^2 \\ -\frac{A_1E_1}{L_1}C_1S_1 & -\frac{A_1E_1}{L_1}S_1^2 & -\frac{A_1E_1}{L_1}C_1S_1+\frac{A_2E_2}{L_2}C_2S_2 \\ \mathbf{0} & \mathbf{0} & -\frac{A_2E_2}{L_2}C_2^2 \\ \mathbf{0} & \mathbf{0} & -\frac{A_2E_2}{L_2}C_2S_2 \end{matrix}\right. \cdots$$

$$\cdots \left.\begin{matrix} -\frac{A_1E_1}{L_1}C_1S_1 & \mathbf{0} & \mathbf{0} \\ -\frac{A_1E_1}{L_1}S_1^2 & \mathbf{0} & \mathbf{0} \\ -\frac{A_1E_1}{L_1}C_1S_1+\frac{A_2E_2}{L_2}C_2S_2 & -\frac{A_2E_2}{L_2}C_2^2 & -\frac{A_2E_2}{L_2}C_2S_2 \\ \frac{A_1E_1}{L_1}S_1^2+\frac{A_2E_2}{L_2}S_2^2 & -\frac{A_2E_2}{L_2}C_2S_2 & -\frac{A_2E_2}{L_2}S_2^2 \\ -\frac{A_2E_2}{L_2}C_2S_2 & \frac{A_2E_2}{L_2}C_2^2 & -\frac{A_2E_2}{L_2}C_2S_2 \\ -\frac{A_2E_2}{L_2}S_2^2 & -\frac{A_2E_2}{L_2}C_2S_2 & \frac{A_2E_2}{L_2}S_2^2 \end{matrix}\right]$$

Then Global Stiffness Matrix for truss elements in parallel:

K =

$$\left[\begin{matrix} \frac{A_1E_1}{L_1}C_1^2+\frac{A_2E_2}{L_2}C_2^2 & \frac{A_1E_1}{L_1}C_1S_1+\frac{A_2E_2}{L_2}C_2S_2 \\ \frac{A_1E_1}{L_1}C_1S_1+\frac{A_2E_2}{L_2}C_2S_2 & \frac{A_1E_1}{L_1}S_1^2+\frac{A_2E_2}{L_2}S_2^2 \\ -(\frac{A_1E_1}{L_1}C_1^2+\frac{A_2E_2}{L_2}C_2^2) & -(\frac{A_1E_1}{L_1}C_1S_1+\frac{A_2E_2}{L_2}C_2S_2) \\ -(\frac{A_1E_1}{L_1}C_1S_1+\frac{A_2E_2}{L_2}C_2S_2) & -(\frac{A_1E_1}{L_1}S_1^2+\frac{A_2E_2}{L_2}S_2^2) \end{matrix}\right. \cdots$$

$$\cdots \left.\begin{matrix} -(\frac{A_1E_1}{L_1}C_1^2+\frac{A_2E_2}{L_2}C_2^2) & -(\frac{A_1E_1}{L_1}C_1S_1+\frac{A_2E_2}{L_2}C_2S_2) \\ -(\frac{A_1E_1}{L_1}C_1S_1+\frac{A_2E_2}{L_2}C_2S_2) & -(\frac{A_1E_1}{L_1}S_1^2+\frac{A_2E_2}{L_2}S_2^2) \\ \frac{A_1E_1}{L_1}C_1^2+\frac{A_2E_2}{L_2}C_2^2 & -(\frac{A_1E_1}{L_1}C_1S_1+\frac{A_2E_2}{L_2}C_2S_2) \\ -(\frac{A_1E_1}{L_1}C_1S_1+\frac{A_2E_2}{L_2}C_2S_2) & \frac{A_1E_1}{L_1}S_1^2+\frac{A_2E_2}{L_2}S_2^2 \end{matrix}\right]$$

Frame:

A rigid plane frame is defined here as a series of beam elements rigidly connected to each other; that is, the original angles made between elements at their joints remain unchanged after the deformation due to applied loads or applied displacements [20].

Stiffness matrix for frame is given:

k =

$$\left[\begin{matrix} \frac{E}{L}(A\cos\theta^2+\frac{12I}{L^2}\sin\theta^2) & \frac{E}{L}(A-\frac{12I}{L^2})\cos\theta\sin\theta & -\frac{6EI}{L^2}\sin\theta \\ \frac{E}{L}(A-\frac{12I}{L^2})\cos\theta\sin\theta & \frac{E}{L}(A\sin\theta^2+\frac{12I}{L^2}\cos\theta^2) & \frac{6EI}{L^2}\cos\theta \\ -\frac{6EI}{L^2}\sin\theta & \frac{6EI}{L^2}\cos\theta & \frac{4EI}{L} \\ -\frac{E}{L}(A\cos\theta^2+\frac{12I}{L^2}\sin\theta^2) & -\frac{E}{L}(A-\frac{12I}{L^2})\cos\theta\sin\theta & \frac{6EI}{L^2}\sin\theta \\ -\frac{E}{L}(A-\frac{12I}{L^2})\cos\theta\sin\theta & -\frac{E}{L}(A\sin\theta^2+\frac{12I}{L^2}\cos\theta^2) & -\frac{6EI}{L^2}\cos\theta \\ -\frac{6EI}{L^2}\sin\theta & \frac{6EI}{L^2}\cos\theta & \frac{2EI}{L} \end{matrix}\right. \cdots$$

$$\cdots\left.\begin{matrix} -\frac{E}{L}(A\cos\theta^2+\frac{12I}{L^2}\sin\theta^2) & -\frac{E}{L}(A-\frac{12I}{L^2})\cos\theta\sin\theta & -\frac{6EI}{L^2}\sin\theta \\ -\frac{E}{L}(A-\frac{12I}{L^2})\cos\theta\sin\theta & -\frac{E}{L}(A\sin\theta^2+\frac{12I}{L^2}\cos\theta^2) & \frac{6EI}{L^2}\cos\theta \\ \frac{6EI}{L^2}\sin\theta & -\frac{6EI}{L^2}\cos\theta & \frac{2EI}{L} \\ \frac{E}{L}(A\cos\theta^2+\frac{12I}{L^2}\sin\theta^2) & \frac{E}{L}(A-\frac{12I}{L^2})\cos\theta\sin\theta & \frac{6EI}{L^2}\sin\theta \\ \frac{E}{L}(A-\frac{12I}{L^2})\cos\theta\sin\theta & \frac{E}{L}(A\sin\theta^2+\frac{12I}{L^2}\cos\theta^2) & -\frac{6EI}{L^2}\cos\theta \\ \frac{6EI}{L^2}\sin\theta & -\frac{6EI}{L^2}\cos\theta & \frac{4EI}{L} \end{matrix}\right]$$

Global Stiffness Matrix for spring elements in series:

Let

cos⁡〖θ_1 〗 = C_1 , cos⁡〖θ_2 〗 = C_2 , sin⁡〖θ_1 〗 = S_1 , sin⁡〖θ_1 〗 = S_2

Consider stiffness matrix for element 1 as

k₁ =

$$\left[\begin{matrix} \frac{E_1}{L_1}(A_1{C_1}^2+\frac{12I_1}{{L_1}^2}{S_1}^2) & \frac{E_1}{L_1}(A_1-\frac{12I_1}{{L_1}^2})C_1S_1 & -\frac{6E_1I_1}{{L_1}^2}S_1 \\ \frac{E_1}{L_1}(A_1-\frac{12I_1}{{L_1}^2})C_1S_1 & \frac{E_1}{L_1}(A_1{S_1}^2+\frac{12I_1}{{L_1}^2}{C_1}^2) & \frac{6E_1I_1}{{L_1}^2}C_1 \\ -\frac{6E_1I_1}{{L_1}^2}S_1 & \frac{6E_1I_1}{{L_1}^2}C_1 & \frac{4E_1I_1}{L_1} \\ -\frac{E_1}{L_1}(A_1{C_1}^2+\frac{12I_1}{{L_1}^2}{S_1}^2) & -\frac{E_1}{L_1}(A_1-\frac{12I_1}{{L_1}^2})C_1S_1 & \frac{6E_1I_1}{{L_1}^2}S_1 \\ -\frac{E_1}{L_1}(A_1-\frac{12I_1}{{L_1}^2})C_1S_1 & -\frac{E_1}{L_1}(A_1{S_1}^2+\frac{12I_1}{{L_1}^2}{C_1}^2) & -\frac{6E_1I_1}{{L_1}^2}C_1 \\ -\frac{6E_1I_1}{{L_1}^2}S_1 & \frac{6E_1I_1}{{L_1}^2}C_1 & \frac{2E_1I_1}{L_1} \end{matrix}\right.\cdots$$

$$\cdots\left.\begin{matrix} -\frac{E_1}{L_1}(A_1{C_1}^2+\frac{12I_1}{{L_1}^2}{S_1}^2) & -\frac{E_1}{L_1}(A_1-\frac{12I_1}{{L_1}^2})C_1S_1 & -\frac{6E_1I_1}{{L_1}^2}S_1 \\ -\frac{E_1}{L_1}(A_1-\frac{12I_1}{{L_1}^2})C_1S_1 & -\frac{E_1}{L_1}(A_1{S_1}^2+\frac{12I_1}{{L_1}^2}{C_1}^2) & \frac{6E_1I_1}{{L_1}^2}C_1 \\ \frac{6E_1I_1}{{L_1}^2}S_1 & -\frac{6E_1I_1}{{L_1}^2}C_1 & \frac{2E_1I_1}{L_1} \\ \frac{E_1}{L_1}(A_1{C_1}^2+\frac{12I_1}{{L_1}^2}{S_1}^2) & \frac{E_1}{L_1}(A_1-\frac{12I_1}{{L_1}^2})C_1S_1 & \frac{6E_1I_1}{{L_1}^2}S_1 \\ \frac{E_1}{L_1}(A_1-\frac{12I_1}{{L_1}^2})C_1S_1 & \frac{E_1}{L_1}(A_1{S_1}^2+\frac{12I_1}{{L_1}^2}{C_1}^2) & -\frac{6E_1I_1}{{L_1}^2}C_1 \\ \frac{6E_1I_1}{{L_1}^2}S_1 & -\frac{6E_1I_1}{{L_1}^2}C_1 & \frac{4E_1I_1}{L_1} \end{matrix}\right]$$

Consider stiffness matrix for element 1 as

k2 =

$$\left[\begin{matrix} \frac{E_2}{L_2}(A_2{C_2}^2+\frac{12I_2}{{L_2}^2}{S_2}^2) & \frac{E_2}{L_2}(A_2-\frac{12I_2}{{L_2}^2})C_2S_2 & -\frac{6E_2I_2}{{L_2}^2}S_2 \\ \frac{E_2}{L_2}(A_2-\frac{12I_2}{{L_2}^2})C_2S_2 & \frac{E_2}{L_2}(A_2{S_2}^2+\frac{12I_2}{{L_2}^2}{C_2}^2) & \frac{6E_2I_2}{{L_2}^2}C_2 \\ -\frac{6E_2I_2}{{L_2}^2}S_2 & \frac{6E_2I_2}{{L_2}^2}C_2 & \frac{4E_2I_2}{L_2} \\ -\frac{E_2}{L_2}(A_2{C_2}^2+\frac{12I_2}{{L_2}^2}{S_2}^2) & -\frac{E_2}{L_2}(A_2-\frac{12I_2}{{L_2}^2})C_2S_2 & \frac{6E_2I_2}{{L_2}^2}S_2 \\ -\frac{E_2}{L_2}(A_2-\frac{12I_2}{{L_2}^2})C_2S_2 & -\frac{E_2}{L_2}(A_2{S_2}^2+\frac{12I_2}{{L_2}^2}{C_2}^2) & -\frac{6E_2I_2}{{L_2}^2}C_2 \\ -\frac{6E_2I_2}{{L_2}^2}S_2 & \frac{6E_2I_2}{{L_2}^2}C_2 & \frac{2E_2I_2}{L_2} \end{matrix}\quad \cdots\right.$$

$$\left.\cdots\quad\begin{matrix} -\frac{E_2}{L_2}(A_2{C_2}^2+\frac{12I_2}{{L_2}^2}{S_2}^2) & -\frac{E_2}{L_2}(A_2-\frac{12I_2}{{L_2}^2})C_2S_2 & -\frac{6E_2I_2}{{L_2}^2}S_2 \\ -\frac{E_2}{L_2}(A_2-\frac{12I_2}{{L_2}^2})C_2S_2 & -\frac{E_2}{L_2}(A_2{S_2}^2+\frac{12I_2}{{L_2}^2}{C_2}^2) & \frac{6E_2I_2}{{L_2}^2}C_2 \\ \frac{6E_2I_2}{{L_2}^2}S_2 & -\frac{6E_2I_2}{{L_2}^2}C_2 & \frac{2E_2I_2}{L_2} \\ \frac{E_2}{L_2}(A_2{C_2}^2+\frac{12I_2}{{L_2}^2}{S_2}^2) & \frac{E_2}{L_2}(A_2-\frac{12I_2}{{L_2}^2})C_2S_2 & \frac{6E_2I_2}{{L_2}^2}S_2 \\ \frac{E_2}{L_2}(A_2-\frac{12I_2}{{L_2}^2})C_2S_2 & \frac{E_2}{L_2}(A_2{S_2}^2+\frac{12I_2}{{L_2}^2}{C_2}^2) & -\frac{6E_2I_2}{{L_2}^2}C_2 \\ \frac{6E_2I_2}{{L_2}^2}S_2 & -\frac{6E_2I_2}{{L_2}^2}C_2 & \frac{4E_2I_2}{L_2} \end{matrix}\right]$$

Then Global Stiffness Matrix for truss elements in series:

Let

$k_{11i} = \frac{E_1}{L_1}(A_1{C_1}^2+\frac{12I_1}{{L_1}^2}{S_1}^2)$ $k_{21i} = \frac{E_2}{L_2}(A_2{C_2}^2+\frac{12I_2}{{L_2}^2}{S_2}^2)$

$k_{12i} = \frac{E_1}{L_1}(A_1-\frac{12I_1}{{L_1}^2})C_1S_1$ $k_{22i} = \frac{E_2}{L_2}(A_2-\frac{12I_2}{{L_2}^2})C_2S_2$

$k_{13i} = \frac{6E_1I_1}{{L_1}^2}S_1$ $k_{23i} = \frac{6E_2I_2}{{L_2}^2}S_2$

$k_{14i} = \frac{E_1}{L_1}(A_1{S_1}^2+\frac{12I_1}{{L_1}^2}{C_1}^2)$ $k_{24i} = \frac{E_2}{L_2}(A_2{S_2}^2+\frac{12I_2}{{L_2}^2}{C_2}^2)$

$k_{15i} = \frac{6E_1I_1}{{L_1}^2}C_1$ $k_{25i} = \frac{6E_2I_2}{{L_2}^2}C_2$

$k_{16i} = \frac{4E_1I_1}{L_1}$ $k_{26i} = \frac{4E_2I_2}{L_2}$

$k_{17i} = \frac{2E_1I_1}{L_1}$ $k_{27i} = \frac{2E_2I_2}{L_2}$

Then Global Stiffness Matrix for frame elements in series:
K=

$$\left[\begin{matrix} k_{11i} & k_{12i} & -k_{13i} \\ k_{12i} & k_{14i} & k_{15i} \\ -k_{13i} & k_{15i} & k_{16i} \\ -k_{11i} & -k_{12i} & k_{13i} \\ -k_{12i} & -k_{14i} & -k_{15i} \\ k_{13i} & k_{15i} & k_{17i} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{matrix}\quad\cdots\right.$$

$$\begin{array}{ccc}
-k_{11i} & -k_{12i} & k_{13i} \\
-k_{12i} & -k_{14i} & -k_{15i} \\
k_{13i} & k_{15i} & k_{17i} \\
(k_{11i}+k_{21i}) & (k_{12i}+\ k_{22i}) & (k_{17i}-k_{23i}) \\
\ldots(k_{12i}+k_{22i}) & (k_{14i}+k_{24i}) & (k_{15i}+k_{25i})\ldots \\
(k_{17i}-k_{23i}) & (k_{15i}+k_{25i}) & (k_{16i}+k_{26i}) \\
-k_{21i} & -\ k_{22i} & k_{23i} \\
-\ k_{22i} & -k_{24i} & -k_{25i} \\
k_{23i} & k_{25i} & k_{27i}
\end{array}$$

$$\left.\begin{array}{ccc}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
-k_{21i} & -\ k_{22i} & k_{23i} \\
\ldots-\ k_{22i} & -k_{24i} & k_{25i} \\
k_{23i} & -k_{25i} & k_{27i} \\
k_{21i} & k_{22i} & k_{27i} \\
k_{22i} & k_{24i} & k_{25i} \\
k_{27i} & k_{25i} & k_{26i}
\end{array}\right]$$

Then Global Stiffness Matrix for frame elements in parallel:
K=

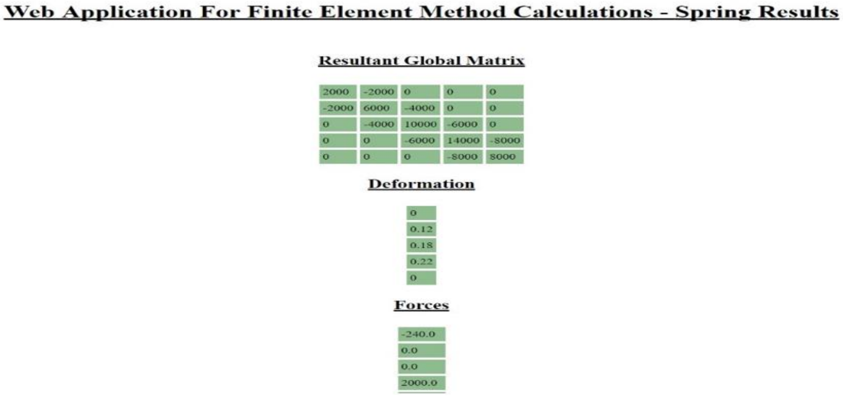
$$\left[\begin{array}{ccc}
k_{11i}+k_{21i} & k_{12i}+\ k_{22i} & -(k_{13i}+k_{23i}) \\
k_{12i}+\ k_{22i} & k_{14i}+k_{24i} & k_{15i}+k_{25i} \\
-(k_{13i}+k_{23i}) & k_{15i}+k_{25i} & k_{16i}+k_{26i} \\
-(k_{11i}+k_{21i}) & -(k_{12i}+\ k_{22i}) & k_{13i}+k_{23i} \\
-(k_{12i}+\ k_{22i}) & -(k_{14i}+k_{24i}) & -(k_{15i}+k_{25i}) \\
(k_{13i}+k_{23i}) & k_{15i}+k_{25i} & k_{17i}+k_{27i}
\end{array}\right.\cdots$$

$$\cdot\cdot\left.\begin{array}{ccc}
-(k_{11i}+k_{21i}) & -(k_{12i}+\ k_{22i}) & (k_{13i}+k_{23i}) \\
-(k_{12i}+k_{22i}) & -(k_{14i}+k_{24i}) & k_{15i}+k_{25i} \\
k_{13i}+k_{23i} & -(k_{15i}+\ k_{25i}) & k_{17i}+k_{27i} \\
(k_{11i}+k_{21i}) & (k_{12i}+k_{22i}) & k_{17i}+k_{27i} \\
(k_{12i}+k_{22i}) & (k_{14i}+k_{24i}) & k_{15i}+k_{25i} \\
k_{17i}+k_{27i} & k_{15i}+k_{25i} & k_{16i}+k_{26i}
\end{array}\right]$$

RESULTS

Illustrative Example #1 Springs

For the given assembly of spring determining global stiffness matrix, displacement and forces. Consider element 1 between node 1 and 2. Element 2 between 2 and 3, element 3 between node 3 and 4 and element 4 between node 4 and 5. Spring constants for each element are k1 = 2000 lb./in, k2 = 4000 lb./in, k3 = 6000 lb./in, k4 = 8000 lb./in respectively. Node 4 is acted upon by a force of 2000 lb.



Web Application For Finite Element Method Calculations - Spring Results

Resultant Global Matrix

2000	-2000	0	0	0
-2000	6000	-4000	0	0
0	-4000	10000	-6000	0
0	0	-6000	14000	-8000
0	0	0	-8000	8000

Deformation

0
0.12
0.18
0.22
0

Forces

-240.0
0.0
0.0
2000.0

Figure 14 - *Spring Calculation Results*

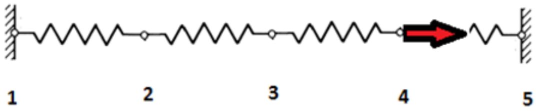


Figure 15 *-Spring Problem*

From the figure boundary conditions can be estimated as given below:
d1 = 0
d5 = 0 as both nodes are fix.
F4 = 2000lb
The unknowns are:
Deformations: d2, d3, d4
Forces: F1, F2, F3, F5.

Manual Calculations:

Upon performing manual calculations results obtained are as follows:
d1 = 0, d2 = 0.12, d3 = 0.18, d4 =0.22, d5 = 0
f1 = -240, f2 =0, f3 = 0, f4 = 2000, f5 = -1760

Web Based Application Calculations:

Inputting the values of stiffness constant for each element as well as inputting the boundary conditions, given forces and total number of nodes in the software we get resultant solution from software as:

Illustrative Example #2 Bars

For the given assembly of bars determining global stiffness matrix, displacement and forces. Consider element 1 between node 1 and 2. Element 2 between 2 and 3 and element 3 between node 3 and 4. Each element length and element area are 17 in and 1 in2 respectively. Considering E = 17 x 10^6 psi for all elements of bar assembly. A force of 2000 lb. is applied at node 3 in positive x direction.

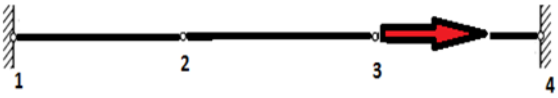


Figure 16 *- Bar Problem*

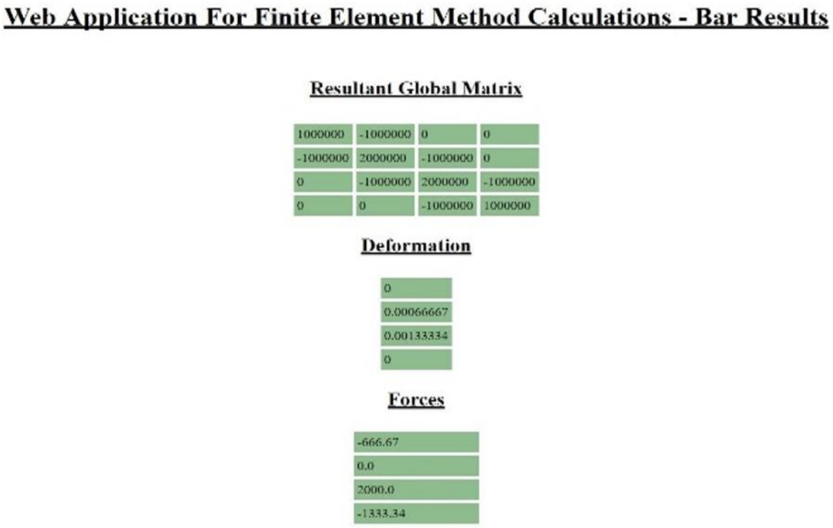
From the figure boundary conditions can be estimated as given below:
d1 = 0
d4 = 0 as both nodes are fix.
F3 = 2000lb
The unknowns are:
Deformations: d2, d3
Forces: F1, F2, F4

Manual Calculations:

Upon performing manual calculations results obtained are as follows:
d1 = 0, d2 = 0.00066667, d3 = 0.00133334, d4 = 0
f1 = -666.67, f2 = 0, f3 = 2000, f4 = 1333.34

Web Based Application Calculations:

Inputting the values of stiffness constant for each element as well as inputting the boundary conditions, given forces and total number of nodes in the software we get resultant solution from software as:



Web Application For Finite Element Method Calculations - Bar Results

Resultant Global Matrix

1000000	-1000000	0	0
-1000000	2000000	-1000000	0
0	-1000000	2000000	-1000000
0	0	-1000000	1000000

Deformation

0
0.00066667
0.00133334
0

Forces

-666.67
0.0
2000.0
-1333.34

Figure 17 *- Bar Calculation Results*

Illustrative Example #3 Beams

For the given assembly of beams determining global stiffness matrix, displacement, rotation, forces and bending moments. Consider beam element 1 between node 1 and 2 and beam element 2 between 2 and 3. Each beam element length is 2 inch. Considering E = 8 x 10^6 psi for all elements of beam assembly. A force of 2000 lb. is applied at node 2 in negative y direction. For all beam elements I = 1 x 10^-4 inch^4.

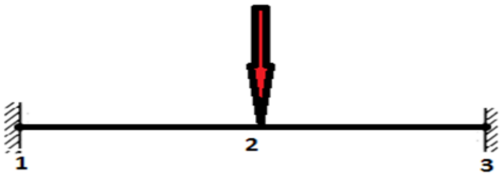


Figure 18 *- Beam Problem*

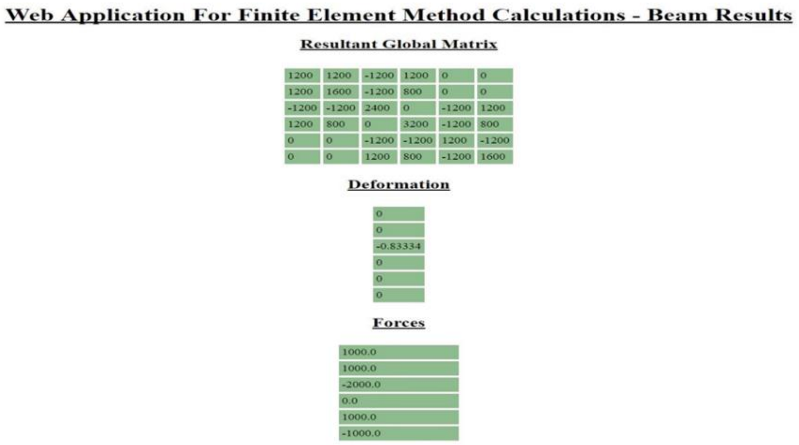
From the figure boundary conditions can be estimated as given below:
d1 = 0
$_{\varphi}$1 = 0
d3y= 0
$_{\varphi}$3 = 0 as both nodes 1 and 3 are fixed.
F2 = -2000lb
The unknowns are:
Deformations: d2, $_{\varphi}$2
Forces: F1, M1, F3, M3

Manual Calculations:

Upon performing manual calculations results obtained are as follows:
d1 = 0, $_{\varphi}$1 = 0, d2 = -0.83334, $_{\varphi}$2 = 0, d3 = 0, $_{\varphi}$3 = 0
F1 = 1000, M1 = 1000, F2 = -2000, M2 = 0, F3 = 1000, M3 = -1000

Web Based Application Calculations:

Inputting the values of stiffness constant for each element as well as inputting the boundary conditions, given forces and total number of nodes in the software we get resultant solution from software as:



Web Application For Finite Element Method Calculations - Beam Results

Resultant Global Matrix

1200	1200	-1200	1200	0	0
1200	1600	-1200	800	0	0
-1200	-1200	2400	0	-1200	1200
1200	800	0	3200	-1200	800
0	0	-1200	-1200	1200	-1200
0	0	1200	800	-1200	1600

Deformation

0
0
-0.83334
0
0
0

Forces

1000.0
1000.0
-2000.0
0.0
1000.0
-1000.0

Figure 19 *- Beam Calculation Results*

Illustrative Example #4 Truss

For the given plane truss, which consist of three elements determining global stiffness matrix, displacement, rotation, forces and bending moments. Consider element 1 between node 1 and 3, element 2 between 1 and 2 and element 3 between node 1 and node 4. Angle between element 1 and element 2 is 300 and angle between element 2 and element 3 is 450. Considering E = 15 x 106 psi for all elements of assembly. A force of 2000 lb. is applied at node 1 in negative y direction. For all elements A = 1 x 10-2 in2 and length of each elements is 15 in.

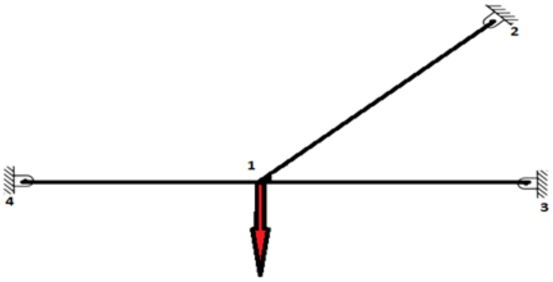


Figure 20 *- Truss Problem*

Figure 20: Truss Problem

From the figure boundary conditions can be estimated as given below:

d2x = 0, d2y = 0, d4x = 0, d4y = 0, d3x = 0, d3y = 0

as nodes 2, 3 and 4 are fixed.

F1 = -2000lb

The unknowns are:

Deformations: d1x, d1y

Forces: F2x, F2y, F3x, F3y, F4x, F4y

Manual Calculations:

Upon performing manual calculations results obtained are as follow

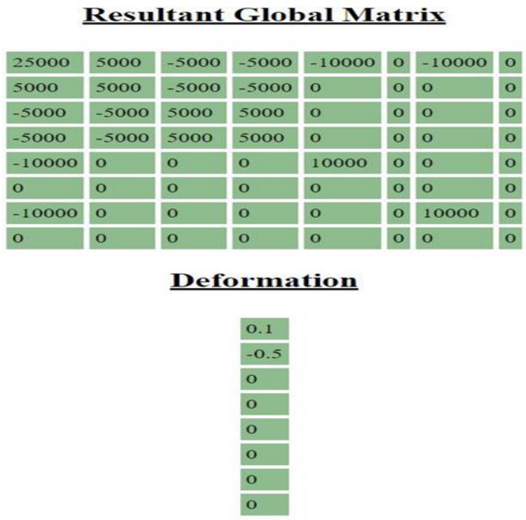
d1x = 0.1, d1y = -0.5, d2x = 0, d2y = 0, d3x = 0, d3y =0

F1x = 0, F1y =-2000 F2x = 2000, F2y = 2000

F3x = -1000, F3y = 0, F4x = -1000, F4y = 0

Web Based Application Calculations:

Inputting the values of stiffness constant for each element as well as inputting the boundary conditions, given forces and total number of nodes in the software we get resultant solution from software as:



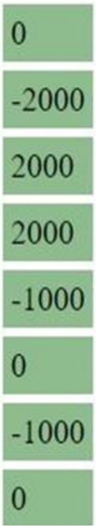
Resultant Global Matrix

25000	5000	-5000	-5000	-10000	0	-10000	0
5000	5000	-5000	-5000	0	0	0	0
-5000	-5000	5000	5000	0	0	0	0
-5000	-5000	5000	5000	0	0	0	0
-10000	0	0	0	10000	0	0	0
0	0	0	0	0	0	0	0
-10000	0	0	0	0	0	10000	0
0	0	0	0	0	0	0	0

Deformation

0.1
-0.5
0
0
0
0
0
0

***Figure 21** - Truss Calculation Results*



0
-2000
2000
2000
-1000
0
-1000
0

***Figure 22** - Truss Calculation Forces*

Illustrative Example #5 Frame

For the given frame, which consist of two elements determining global stiffness matrix, displacement, rotation, forces and bending moments. Consider element 1 between node 1 and 2, element 2 between 1 and 3. Angle between element 1 and element 2 is 450. Considering E = 6 x 10^6 psi for all elements of assembly. A force of 2000 lb. is applied at node 1 in positive x direction. For all elements I = 3 x 10-4 in2 and length of each elements is 6 in.

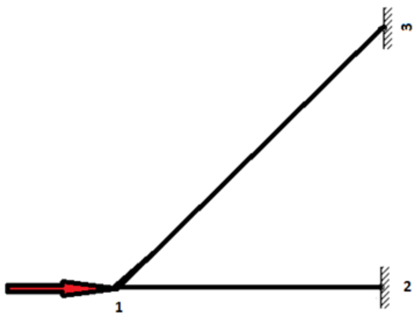
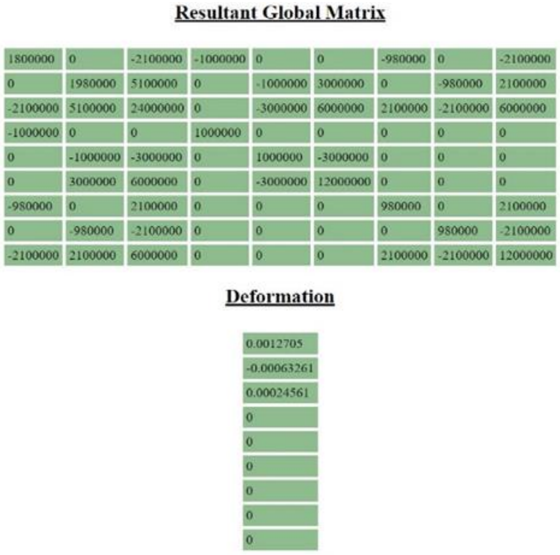


***Figure 23** - Frame Problem*

From the figure boundary conditions can be estimated as given below:
d2x =d2y= $_{\varphi}$2 = d3x=d3y= $_{\varphi}$3 = 0

as nodes 2 and 3 are fixed.
F1 = 2000 lb



Resultant Global Matrix

1800000	0	-2100000	-1000000	0	0	-980000	0	-2100000
0	1980000	5100000	0	-1000000	3000000	0	-980000	2100000
-2100000	5100000	24000000	0	-3000000	6000000	2100000	-2100000	6000000
-1000000	0	0	1000000	0	0	0	0	0
0	-1000000	-3000000	0	1000000	-3000000	0	0	0
0	3000000	6000000	0	-3000000	12000000	0	0	0
-980000	0	2100000	0	0	0	980000	0	2100000
0	-980000	-2100000	0	0	0	0	980000	-2100000
-2100000	2100000	6000000	0	0	0	2100000	-2100000	12000000

Deformation

0.0012705
-0.00063261
0.00024561
0
0
0
0
0
0

***Figure 24** - Frame Calculation Results*

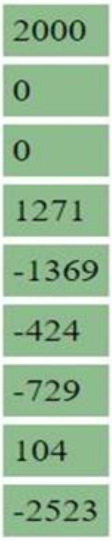
Deformations: d1x, d1y, φ1
Forces: F2x, F2y, M2, F3x, F3y, M3

Manual Calculations:

Upon performing manual calculations results obtained are as follow
d1x = 0.0012705, d1y = -0.00063261, φ1 = 0.00024561,
d2x = 0, d2y = 0, φ2 = 0,
d3x = 0, d3y =0, φ =0
F1x = 2000, F1y = 0, M1 = 0,
F2x = 1270.6, F2y =-1369.44, M2 = -424.17,
F3x = -729.428, F3y = 104.17, M3 = -2523.081

Web Based Application Calculations:

Inputting the values of stiffness constant for each element as well as inputting the boundary conditions, given forces and total number of nodes in the software we get resultant solution from software as:



2000
0
0
1271
-1369
-424
-729
104
-2523

***Figure 25** - Frame Calculation Forces*

Comparison With Existing Software

As can be seen in table 4, WAFEC and four other different types of tools were used to calculate the time required to complete sample HW problems. Please note that table 4 is not intended to compare and contrast the different software's as each of these software's are proficient in its own special settings.

Table 4 – Average time required to complete sample academic HW problems (among 6 students)

#	Software	Time elapsed to complete for each element type (min)				
		Spring	Truss	Bar	Frame	Beam
1	5.2	6.8	5.8	7.3	8.1	5.2
2	24.3	26.4	24.9	26.8	29.3	24.3
3	15.2	15.8	14.8	16.5	16.8	15.2
4	15.4	15.3	16.2	15.9	16.9	15.4
5	28.6	30.1	32.1	32.0	29.8	28.6

This exercise was intended more as a mechanism to estimate the time required to complete academic finite element HW problems that otherwise would take a substantial amount of time when done by hand calculations. The time calculated was based on a start point of opening the software to setting it up with the initial and boundary conditions, inputting the requisite data, to finally receiving the results. Since WAFEC had an inherent advantage of having the code set up especially for academic FEA HW problem solving, it had an advantage over other software's, some of which are numerical problem-solving based like Ansys and Abaqus, and others are analytical based like MS Excel. We understand and appreciate the fact that these other software's have created their own niche in various applications. The elapsed time from inception to completion that is reported on table 4 is an average from the time taken for six students (all having similar academic backgrounds), done independently. As evident from table 4, WAFEC gave a clear and distinct advantage over other software's by taking only between 1/3rd and 1/6th of the time required for other software's. This will reduce among other things, costs, and increase efficacy of the entire exercise.

CONCLUSIONS

The objective of this paper is to describe the application of a web-based software WAFEC in solving finite element problems related to mechanical and structural engineering. Types of finite elements problems were limited to one dimensional, two dimensional and three-dimensional elements. Java / J2ee language is used as default programming technologies for WAFEC due to its strength in portability, flexibility and ability to perform computations. Defining finite element model is done by inputting information in JSP Pages which is taken as input by servlets of WAFEC and performs computations to generate results.

Creation of WAFEC as web-based software eliminates the limitations of desktop applications and increases flexibility. Having a web-based environment allows users to understand, solve and analyze finite element problems with greater control. It acts as a teaching aid and helps to cross check solution and solve academic text book problems. Future Programs may include addition of different type of elements and different materials and integration with design packages.

It can be concluded from the research that:

- Creation of WAFEC as web-based software eliminates the limitations of desktop applications and increases flexibility.
- It is very portable compared to other commercial FEM packages.
- WAFEC is more time efficient than other techniques of solving problems especially using manual approach and excel solving techniques.
- Based on above results it can be inferred that as number of elements and complexity increases the percentage of time saving using WAFEC increases.
- It is simple, easy to use and user friendly.
- It is more academic oriented rather than industrial oriented as compare to other commercial FEA packages.
- Readability of output is better compare to several commercial FEM packages.
- Having a web based environment allows users to understand, solve and analyze finite element problems with reater control.
- It acts as a teaching aid and helps to cross check solution and solve academic text book problems.

Future Programs may include addition of different type of elements and different materials and integration with design packages.--

REFERENCES

[1]. Kumar, S. (2010). Object-oriented finite element programming for engineering analysis in c++. Journal of Software, 5(7), 689-696.

[2]. Siswanto, W. A., & Darmawan, A. S. (2012). Teaching finite element method of structural line elements assisted by open source freeMat. Research Journal of Applied Sciences, Engineering and Technology, 4(10), 1277-1286.

[3]. Gunakala, S. R., Comissiong, D. M. G., Jordan, K., & Sankar, A. (2012). A Finite Element Solution of the Beam Equation via MATLAB. International Journal of Applied, 2(8).

[4]. Neehaar, D. (2013) Modal Analysis Of A Stepped Bar Using Matlab & Ansys. International Journal of Engineering and Application, 3(2), 504-509.

[5]. Zienkiewicz OC, Taylor RL and Zhu JZ. (2015) The finite element method: its basis and fundamentals, 6th ed. Oxford, UK: Butterworth-Heinemann.

[6]. Cook, R. D., Malkus, D. S., Plesha, M. E., and Witt, R. J. (2002) Concepts and Applications of Finite Element Analysis, 4th ed., Wiley, New York.

[7]. Bathe KJ. (1996) Finite element procedure. New Jersey, USA: Prentice-Hall.

[8]. S. C. Brenner and L. R. Scott. (2002) The mathematical theory of finite element methods, volume 15 of Texts in Applied Mathematics. 2nd ed. Springer-Verlag, New York.

[9]. C. Pozrikidis. (2005) Introduction To Finite And Spectral Element Methods Using Matlab. Chapman & Hall/CRC.

[10]. Nikishkov, G. P. (2010). Programming Finite Elements in JavaTM. Springer Science & Business Media.

[11]. X.A. Kong. (1996). Data design approach for object-oriented FEM programs", Comput. Struct., vol. 61, pp. 503-513.

[12]. G. R. Miller. (1991) An Object-Oriented Approach to Structural Analysis and Design. Comput. Struct., vol. 40, pp. 75-82..

[13]. B.C.P. Heng and R.I. Mackie. (2009) Using design patterns in object-oriented finite element programming. Comput. Struct., vol. 87, pp. 952–961.

[14]. L. Yu and A.V. Kumar. (2001) An object-oriented modular framework for implementing the finite element method. Comput. Struct., vol. 79, pp. 919-928.

[15]. T. Zimmermann, Y. Dubois-Pélerin and P. Bomme (1992) Object-oriented Finite Element Programming: II. A Prototype Program in Smalltalk, Computer Methods in Applied Mechanics and Engineering, 98, No.3, pp. 361-397

[16]. Grafton, P. E., & Stome, D. R. (1963). Analysis of axisymmetrical shells by the direct stiffness method. AIAA journal, 1(10), 2342-2347.

[17]. Chessa, J. (2002). Programing the finite element method with Matlab. Northwestern University, 20(02).

[18]. Raj, R., Sinha, P. K., & Prakash, E. V. Modelling, simulation and analysis of cantilever beam of different material by finite element method, ansys & matlab. Governing, 4, 5.

[19]. Zuo, W., Bai, J., & Cheng, F. (2014). EFESTS: Educational finite element software for truss structure. Part I: Preprocess. International Journal of Mechanical Engineering Education, 42(4), 298-306.

[20]. Logan, D., 2007. A First Course in the Finite Element Method Fourth Edition. Thomson Engineering, Canada.

[21]. Hrennikoff, A. (1941) 'Solution of Problems in Elasticity by the Frame Work Method. Journal of Applied Mechanics, Vol. 8, No. 4, pp. 169–175.

[22]. Turner, M. J., Clough, R. W., Martin, H. C., and Topp, L. J. (1956) Stiffness and Deflection Analysis of Complex Structures. Journal of Aeronautical Sciences, Vol. 23, No. 9, pp. 805–824

[23]. Clough, R. W. (1960) The Finite Element Method in Plane Stress Analysis. Proceedings, American Society of Civil Engineers, 2nd Conference on Electronic Computation. pp. 345–378.

[24]. Mathworks, 2011. MATLAB. Retrieved from: http://www.mathworks.com/products/matlab

[25]. Swanson, J. A., ANSYS-Engineering Analysis Systems User's Manual, Swanson Analysis Systems, Inc., Johnson Rd., P.O. Box 65, Houston, PA 15342. (Mathworks, 2011)

[26]. Algor Interactive Systems, 150 Beta Drive, Pittsburgh, PA 15238.

[27]. MSC/NASTRAN, MacNeal-Schwendler Corp., 600 Suffolk St., Lowell, MA, 01854.

[28]. STARDYNE, Research Engineers, Inc., 22700 Savi Ranch Pkwy, Yorba Linda, CA 92687.