Faculty of Engineering, Built Environment and Information Technology

Fakulteit Ingenieurswese, Bou-omgewing en Inligtingtegnologie

School of Engineering

Mechanical Engineering

MKM321 Solid Mechanics

Lecturer: Prof. Nico Wilke

Last Revision: 15 July 2020

1 Departmental Study Guide

This study guide is a crucial part of the general study guide of the Department. In the study guide of the Department , information is given on the mission and vision of the department , general administration and regulations (professionalism and integrity, course related information and formal communication, workshop use and safety, plagiarism, class representative duties, sick test and sick exam guidelines, vacation work, appeal process and adjustment of marks, university regulations, frequently asked questions), ECSA outcomes and ECSA exit level outcomes, ECSA knowledge areas, CDIO, new curriculum and assessment of cognitive levels. It is expected that you are very familiar with the content of the Departmental Study Guide. It is available in English and Afrikaans on the Department's website.
English

`https://www.up.ac.za/media/shared/120/Noticeboard/2021/departmental-studyguide-eng-2021_version27may2021.zp204392.pdf`

Take note of the specific instructions in the above study guide on:

a. Safety

b. Plagiarism

c. What to do if you were sick (very important)?

d. Appeal process on the adjustment of marks

This study guide will be continuously updated throughout the semester, it is the students' responsibility to ensure they are consulting the latest version of the study guide.

2 Educational approach

"Leaning without reflection is a waste. Reflection without learning is dangerous." - Confucius

The educational approach is this course is based on two fundamental pillars:

1. Reflective learning to achieve
2. conceptual understanding

The teaching approach in this course is cemented in reflective teaching, while the learning approach is cemented in reflective learning. Well what does this reflective aspect imply?

2.1 Reflective Teaching

"If a child can't learn the way we teach, maybe we should teach the way they learn." - Ignacio Estrada

Julie Tice eloquently states that Reflective teaching means looking at what we do in the classroom, thinking about why we do it, and thinking about if it works - a process of self-observation and self-evaluation.

As a consequence this course is continuously being developed based on critical assessment of which approaches and delivery styles consistently achieve their outcomes and those that do not. As educator my aim is to ensure that the content and presentation adds as much value as possible.

2.2 Reflective Learning

"We do not learn from experience... we learn from reflecting on experience." - John Dewey

The main educational approach of this course is **Reflective Learning** to ensure **Conceptual Understanding**. Reflective learning highlights the importance of thinking about what we did, why we did it and how it is impacting my current understanding, after we have done something.

In this course, doing a homework problem is not learning and definitely not enough to pass this course, but only the first step in setting up learning to happen. Hence, homework and assessment actively incorporate reflective questions, after something has been done, it is by engaging with these questions where conceptual understanding is developed and learning happens. As you will see the doing part is mechanistic and basic first year Algebra.

Most might have heard the saying similar to this one:

"If you give a person a fish, he will be hungry tomorrow. If you teach a person to fish, he will be richer forever." - Chinese proverb

An equivalent statement about learning is:

"If you teach a person to solve a problem, he will be stuck tomorrow with another problem. If you teach a person the concepts behind a problem, she will be your and your children's teacher one day." - Corny proverb

Elementary example of a mechanistic question:
Consider a mass of 1 kg moving at a constant velocity of 1 m/s. What is the kinetic energy of the body?

$$E_k = \frac{1}{2}J \tag{1}$$

It is elementary to do this homework question, but doing it and obtaining the correct answer says nothing of your understanding of kinetic energy but merely illustrates that you can substitute numbers into a formula and use your calculator correctly.

Elementary example of two conceptual questions:
Given that the kinetic energy of an object moving at constant velocity is $1/2J$, by doubling the velocity of the object, the kinetic energy doubles. Is this statement true or false?

Given that the kinetic energy of an object moving at constant velocity is $\frac{1}{2}J$, by doubling the mass of the object the kinetic energy doubles. Is this statement true or false?

Here, you have some fundamental understanding of kinetic energy by reflecting after having done some homework. Those who appreciate this understanding, would take care when driving at 120 km/h as opposed to 60 km/h.

2.3 Why this approach

"A teacher only delivers in a learner what she/he assesses in a learner." - DNW

To understand the motivation for this approach, it helps to give historical context of historical educational approaches.

Since, teaching MKM321 in 2017, it quickly became evident that students failed to grasp even the most elementary mathematical and structural mechanics concepts that were taught in the structural mechanics curriculum. Although students completed a vast amount of complex homework problems in previous courses, students failed to learn even the most basic structural mechanics concepts. I can confirm that this is a general trend, locally and abroad, through my interactions as external examiner with structural mechanics courses, as well as, through open discussions with my international research collaborators and my jointly supervised students in the USA, UK, France and China.

Essentially, historical education approaches are failing to equip modern students with even the basic fundamental knowledge onto which a structural curriculum could extend. Based on critical thinking and assessments, the reasons I have identified are largely due to the following three factors:

1. excessive homework forced learners to only do homework, and failed to encourage them to stop, think about what they were doing and why they were doing it, i.e. excessive homework indirectly discouraged reflection,
2. homework failed to incorporate reflective questions, i.e. questions that required students to think about what they did, why they did it and what would happen if something changed,
3. assessment focused mainly on imitation, that is mechanistic and formula driven routine work that is aggravated by assessing learners during tests and exams on questions that are identical to class or homework examples as opposed to assessing concepts and conceptual understanding.

Since then I have actively engaged lecturers teaching courses that precede MKM321, and actively canvassed for reflective learning and stressed the importance of ensuring that conceptual understanding is assessed in the structural mechanics stream. Hopefully, you would have encountered multiple-choice questions that assessed your understanding without involving excessive calculation. You should have encountered homework, test and exam questions that for example described a problem with its answer given and asked you to reflect or demonstrate conceptual understanding of the underlying concept. The importance of highlighting this is to remind you of this teaching and assessment style as this course is based on this premise.

2.4 Course Roadmap

The nice thing about this course is that it is ONLY about approximating functions given some data. Yes with approximation of a function I mean, similar to what you did in MPR213, when you fitted a line or exponential curve through some data points in Excel or Python a.k.a. regression. Before I continue, let us reflect on the implication of REGRESSION. Regression is the keystone of Artificial Intelligence, Autonomous Vehicles, Robots, Deep Fakes, Automated Fake News Articles, Statistical Learning, Machine Learning, Deep Learning, Reinforcement Learning, Internet of Things and the Fourth Industrial Revolution - **Yes, these are the things that most people fear about the future, and it is all based on REGRESSION, i.e. fitting functions through data using Python!**. Test yourself against Deep Fakes - which of these two faces in Figure?1, are of an actual person born on planet earth, and which one is entirely fictional, i.e. completely computer generated like a computer game character, i.e. someone that has never been born?

Figure 1: Power of regression - Which of these two faces are of an actual person and which one is computer generated? Read this article to find out: `https://medium.com/syncedreview/gan-2-0-nvidias-hyperrealistic-face-generator-e3439d33ebaf`

The reason I highlight this is that you enter this course with an understanding that, what we do in this course is entirely relevant to what is happening in the "real" world out there, relevant to the latest and greatest scientific breakthroughs of 2020, but more importantly job security for your future as a Mechanical Engineer. The concepts used in this course were in fact used to construct the first image of a black hole that gripped the world in April 2019 (`https://www.youtube.com/watch?v=0RxitCeuokI`). It is therefore not so strange that my personal research has a significant Deep Learning focus, applied to Mechanical Engineering Problems and Modelling of Mechanical Engineering Problems. It is indeed an amazing time to consider postgraduate studies - feel free to talk to me if you are interested!

However, in this course the data that are usually supplied about the problem is very limited. This makes fitting a function through only the supplied data using regression quite senseless. Senseless in that the models are not very useful using only what has been supplied. The consequence of having very limited data is that we need to supply additional information to fit these functions. What additional information do we need to supply? In this course our additional information are physics and mathematical models, in particular models of solid mechanics. Not only has this resulted in Computational Engineering Software of the last 50 years, such as Ansys Fluent, MSC Patran and StarCCM+, but it is a very active international research topic, one in which, Bren Professor at Caltech Computing + Mathematical Sciences Department (`http://tensorlab.cms.`

`caltech.edu/users/anima/`), has made some considerable contributions.

This course is an introduction to the *numerical function approximation approach*, the finite element method, in the field of *solid mechanics*. The majority of the finite element method will be developed for one dimensional problems and then extended by reasoning of the concepts to two dimensional problems. This is important, since extending from 1-D to 2-D forces us to really think critically about the 1-D concepts, essentially, it forces us to have a polished 1-D structural mechanics understanding as opposed to a superficial understanding. This section also demonstrates that by having a polished 1-D structural mechanics understanding, we essentially also get a polished 2-D and 3-D understanding, by merely thinking about the extension towards 2-D and 3-D.

It is essential that learners develop and are able to demonstrate their numerical literacy i.e. ability to reason about an equation, compute on the computer (in this course Python) and sensibly interpret the results. Numerical literacy has become an essential and basic outcome for any engineering course, this course aims to develop this skill further within the context of solid mechanics. Fundamental literacy tips to reason about an equation are:

1. Ask yourself what will happen when you make a variable smaller or bigger, e.g. halving or doubling.

2. Isolate different variables of an equation and repeat the previous step.

3. Apply the equations to things around you. Our biggest asset as Mechanical Engineers is that we work with things that we can observe - all the mathematics and modelling that we do, merely reflects our current understanding of what we see and what we think happens.

The course is divided into three parts. The first part develops the additional information and mathematical language required for us to be able to perform regression:

1. The mathematical language of *solid mechanics* is formulated using *continuum mechanics* (Chapter 1),

2. to define field quantities (spatial functions) such as displacement, traction, strain and stress (Chapter 1),

3. and equilibrium in infinitesimal form (stress equilibrium) as opposed to macroscopic force and moment equilibrium (Chapter 1),

4. which are described by fifteen governing differential and Algebraic equations that solve for fifteen unknown spatial field quantities (spatial functions) (Chapter 1 + Supplementary Notes) known as the strong form of the governing equations.

To develop our ability to compute numerically, we use Python as an advanced calculator to solve symbolic Algebraic equations (Python Notebook) and numerical Algebraic equations (Chapter 2, Python Notebook). Note the prescribed textbook only provides `Fortran` listed code, which is supplemented by the lecturer in the form of Python Notebooks.

Continuum mechanics deals with the motion, deformation and mechanical behaviour of materials which can be modelled as continuous (smooth) media rather than as discrete particles.

Solid Mechanics builds on all the mathematical courses which students have previously taken, as well as on knowledge of structural analysis (MOW227) and programming (MPR213). The theoretical approach will complement existing applied knowledge, and will open the door to further independent learning, as well as, research opportunities in postgraduate studies.

The second part focuses on how we do regression when following the finite element approach, i.e. how to approximate spatial functions with the focus on one-dimensional problems:

1. The numerical approximation of functions will be recapped and the equations for Galerkin's method derived (Chapter 1, 3 and Supplementary Jupyter-Notebooks) i.e. weak form of the governing equations,

2. Starting with single element problems developing the ideas of numerical integration, one dimensional linear and quadratic elements, element stiffness matrix, element node numbering, element degrees of freedom, traction boundary conditions, body forces, consistent nodal loads, element load vector and displacement boundary conditions, solving linear Algebra equations using the elimination approach and recovering reaction forces (Chapter 3 and Supplementary Jupyter-Notebooks).

3. Evolving to two element problems ideas of global stiffness matrix and global load vector, global degrees of freedom, element numbering, global node numbering.

4. Finally given an arbitrary number of elements the ideas and general concepts of a finite element program will be developed, that is, following a systematic problem solving approach which involves a systematic way to define a problem, setup the numerical equations, solve the equations and post-process the results.

5. Important themes such as linear superposition (Chapter 1), Saint Venant's principle (Chapter 1) and linear Algebra (system of equations, rank and eigenvectors) (Chapter 2) will be continuously revisited.

The third part is dispersed throughout the semester, with a strong focus towards the end. This part develops our numerical literacy and ability to interpret results. It forces us to differentiate between three types of answers or expectations:

1. What can I expect based on real life observations of structures?

2. What can I expect from our simplified model of structures? Here, we need to be able to distinguish between what reality says about structures and what our simplified model says about structures, and how they differ, where they differ and why they differ. If we can't do this then we also can't distinguish between when we have computed and answer correctly using our simplified model from when we make a basic computing mistake using our mathematical model.

3. What can I expect from our finite element approximations of our simplified model of structures? Here, we need to be able to distinguish between what our simplified model says

about structures and what our finite element approximations says about structures. How they differ, where they differ and why they differ. If we can't do this then we also can't distinguish between when we have computed and answered correctly using the finite element method from when we have made a basic computing mistake using the finite element.

In addition, this part ensures that learners can extend the concepts developed in the second part to two-dimensions (Chapter 8) by reasoning and extending the concepts, critically interpret the finite element results and apply linear superpositioning (Chapter 1) and Saint Venant's principle (Chapter 1) to properly model finite element problems:

1. Extending concepts like numerical integration, plane stress problems, plane strain problems, displacement field, strain field, stress field, nodal element numbering and degrees of freedom, traction vector, body forces, prescribed displacements, reaction forces towards two-dimensions.

2. Two dimensional linear (four-node quadrilateral) and quadratic (eight-node quadrilateral) elements.

3. Shear locking in the four-node quadrilateral element (Investigation by Jupyter-Notebook).

4. Symmetry, anti-symmetry (Chapter 4) and linear superposition (Chapter 1) in modelling.

5. Saint Venant's principle applied to displacement boundary conditions and consistent nodal loads when modelling (Chapter 1, 8 and Jupyter-Notebooks).

6. Verification i.e. independent estimate of the solution or anticipated solution to the problem (Chapter 8 and NAFEMS V&V Notes).

7. Validation is the model appropriate for the actual problem at hand (Chapter 8 and NAFEMS V&V Notes).

8. Introduction to dynamic problems including the lumped mass matrix, consistent mass matrix, the general eigenvalue problem and symmetry and anti-symmetry revisited.

Since this course involves learning a new mathematical language supplemented by a numerical language Python, it is essential to practice. Consequently, there will be sufficient homework problems available that cover the material covered in class in more detail, with active encouragement on reflection. There will be weekly opportunities in the form of a two hour practical session on Mondays, to consult with lecturers and with teaching assistants to ensure that learners understand the material thoroughly.

2.5 Homework

"By reflecting on experience alone can we learn wisdom - unfortunately, many pretend to be wise by two methods: First, by imitation, which is easiest; and second by experience, which is the bitterest." - Unknown

General premise of learning is to ensure that the learners are sufficiently challenged such that they are out of their comfort zone but not too much that they become completely de-motivated. Learning happens when a learner dedicates a focused 160 hours of concentration, sweat and significant effort to work through initial emotions of frustration and helplessness, to only discover that she/he is empowered to complete the tasks given necessary dedication to extend his/her comfort zone to include what MKM 321 entails.

Essentially, when enrolling at a University you pay someone to challenge you and push you out of your comfort zone. One of the first signs of being outside of your comfort zone is when you feel frustrated i.e. you are paying someone to frustrate you so that you can grow and learn to feel less frustrated. Fortunately, there is always a lecturer or teaching assistant to assist should you start to feel overwhelmed, make sure you use the opportunities! Important to understand that you do the thinking, as it is the only way your brain develops. Getting an answer from someone only supports the development of someone else's brain and brings superficial relief but your ability to think remains unchanged. This is why memoranda are often such a big disaster for most learners. It improves memory, consumes lots of time and brings emotional relief and often a superficial impression of understanding or even mastery, but unfortunately your ability to think remains unchanged when looking at a memo. Critically engaging with a memorandum can bring lots of insight, but you need to do more than read, you need to properly reflect - if you are even slightly unsure, then best be advised to stay away from memos!

Remember that only your memory improves when you memorise what someone else is saying. Your thinking improves when you reflect on what you have done or someone else has said. There is no shortcut for spending time on reflection, it requires active participation and dedication - what you put in, is what you get out!

2.6 Advanced calculator - Python

"Why do Mechanical Engineers need to program? I am not a programmer!
For the same reason you read and write, although you are not Rowling, Angelou or Shakespeare !"

It may seem contradictory that this course, which is a largely based on computing analytical derivatives and numerically integrating structural mechanics equations, i.e. extensive linear Algebra, could be approached this way. How do we avoid all the Calculus differentiation and Algebra equations?

The answer is simple - by upgrading our pocket calculator to our tablet, notebook, laptop or desktop calculator, namely **Python**.

As educator and researcher I had to upgrade my basic primary school pocket calculator numerous times to get to the one I currently use on a daily basis, which is CUDA that executes on graphical processing units (GPUs) or video and gaming cards that are usually used to make games look great. This calculator is used to simulate granular materials that interact with fluids to solve actual engineering problems in industry.

Even though your Python may be rusted or you find it difficult to get going again, it is important to not flatter yourself by thinking that you are busy with programming in Python. We will however be doing lots of computing using Python - trust me programming requires extensive understanding of classes, constructors, encapsulation, inheritance, polymorphism , design patterns to name a few. Most probably none of that makes sense, and believe me, at the end of the course you will be no better equipped to make sense of any of that.

We will however journey into the following modules to make sure we focus on the concepts, while Python takes care of the computations:

1. scipy (Scientific Python - essentially numpy plus more)
2. sympy (Symbolic Python - as the name implies, allows us to integrate, differentiate and plot symbolically instead of numerically)

3 Lecturers, venues and consulting hours

3.1 Lecturers

Prof. Nico Wilke, Module coordinator
Eng. I, room 10-26
Tel: (012) 420-2861
E-mail: `nico.wilke@up.ac.za`

3.2 Teaching assistants

Teaching assistants with contact information will be published once the course begins. Teaching assistants will assist at tutorials, will offer consultation hours.

3.3 Consultation hours

The lecturers and teaching assistants will schedule consultation hours, which will be published both on the class website (ClickUP). All other consultation will be strictly by appointment.

Learners are encouraged to utilize consultation hours as well as tutorial times. Should you have a problem with the mathematics or Python programming associated with this course, or one of the assignment problems, it is important that you first communicate your own attempt to solve the relevant problem to the instructor. This is to allow the instructor to understand what the problem is, but **more importantly why** you are struggling with a specific concept or problem, so that guided and focused assistance can be given.

Communication regarding important issues that concern the class (e.g. work load at a given moment, etc.) should preferably be put to the lecturer by the class representative at the start of a contact session, or alternatively during consulting hours.

4 Study materials and purchases

4.1 Prescribed text

Chandrupatla, T.R. and Belegundu, A.D., *Introduction to Finite Elements in Engineering*, Pearson, 4th edition, 2011.

A previous or more recent edition of the book will also be adequate.

4.2 Prescribed software

Python Anaconda using Python 3.6 will be used in this course and can be freely downloaded following `https://docs.anaconda.com/anaconda/install`.

In particular, the Jupyter notebook interface will be used as it serves as a documentation platform with Python code. A tutorial on Jupyter notebook will be given in class.

4.3 Recommended texts

These books give alternative presentations of the concepts we will be covering in the course and may be useful to some learners. Some of the books are available electronically through the university library.

- Logan, D.L., *A First Course in the Finite Element Method*, CL Engineering, 2011.
- Reddy, J.N., *Principles of Continuum Mechanics*, Cambridge University Press, Cambridge, 2010.[1]
- Sadd, M. H., *Elasticity, Second Edition: Theory, Application and Numerics*, Elsevier, 2009.

As this course aims to lay a foundation for learners to built upon once this course has been completed. It is important to note resources available at the International Association for the Engineering Modelling, Analysis and Simulation Community, which is useful for additional information as well as online courses that cover modelling strategies for specific classes of problems to further your understanding and modelling capability using simulation. NAFEMS also offers Professional Simulation Engineer Certification that is internationally recognized.

5 Learning activities

5.1 Contact time and learning hours

Number of lectures per week: 3 (50 minutes per lecture)
Number of tutorials per week: 1 (110 minutes per tutorial)

[1] This book was the prescribed textbook for MKM321 for the past few years, when continuum mechanics was the focus of the course.

This module carries a weighting of 16 credits, indicating that on average a learner should spend some 160 hours (including time for preparation for tests and examinations) to master the required skills. This means that on average you should devote some 12 hours of study time per week to this module. The scheduled contact time is approximately 5 hours per week, which means that another 7 hours per week of your own study time should be devoted to the module in the form of completing weekly or bi-weekly assignments.

In addition, Structural Design (MOW227) is a prerequisite for this course. It is essential that all learners for MKM321 is comfortable with the content of MOW227. As a reminder please consider the yearbook description for MOW227:

```
Analyse statically determinate structures to obtain section forces and moments
and stress distributions.

Thin-walled pressure vessels.

Stress and strain transformations. Introduction of stress tensor. Derivation of
stress transformation equations. Eigenvalue/vector analysis for principle
stresses and strains.

Mohr''s circle. Failure criteria. Fatigue strength design. All analysis techni-
ques above are applied to the open-ended design of components like beams and
shafts.
```

This course articulates and extends on MOW227 as follows:

1. Analysis of stress fields and displacement fields instead of only at a point in a structure
2. Extension of the traction vector into the stress tensor
3. Prescribed displacement boundary conditions e.g. clamped (displacements in al directions are zero at the point that is clamped), vertical roller (displacement in x-direction is restricted) and horizontal rollers (displacement in y-direction is restricted) and prescribed displacement (displacement at a point is prescribed).
4. Analysis of statically indeterminate structures in general i.e. additional equations to the macroscopic equilibrium is required to solve the problem i.e. knowledge of the material of the structure is essential to solve problems in this course.
5. Infinitesimal equations for equilibrium instead of the macroscopic equilibrium using sum of forces equal zero along the Cartesian directions or sum of moments is zero around any point along the Cartesian axes.

5.2 Homework Assignments

Homework assignments will be given every week to two weeks on ClickUP, usually on a Friday, and will be due on the Thursday the day before the new assignment is posted. Assignments need to be uploaded under the appropriate assignment on ClickUP as a **Jupyter-notebook**. A grade will be awarded based on effort, insight and intepretation of the results when conducting the assignments, not the correctness of the assignment to encourage extensive reflection. This will form part of the continuous assessment in this course.

It is important to note that almost all assignments will involve Python programming, with Python being utilized such that the majority of the algebraic manipulations being done in Python to alleviate the student from tedious manipulations. The aim of this is to allow the student to focus on the concepts, to visually explore the mathematics, critically discuss the implication of a calculation and critical interpret the obtained results. In short, completing the calculation for an assignment is merely the start of the assignment. As a consequence the ability to conduct calculations in absence of understanding as to the concept behind the calculation, the implication of a calculation or critical interpretation of an obtained result would not be sufficient to obtain a pass for this course.

It is also important to note that the assignments will follow a numerical investigation approach that build on each other. The implication is that not doing an assignment will adversely affect the assignments to follow.

Selected solution memoranda will be published on ClickUP.

Collaborating with peers on difficult problems, consulting with instructors and teaching assistants, and explaining the concepts as you understand them will all contribute to your success in the course. Although we strongly encourage you to work with each other while you are learning, remember that to pass the course, each of you must be able to solve the problems independently. The skills and knowledge developed through completing homework assignments will be assessed in tutorial tests, in-class quizzes as well as the semester tests and final exam. The effort you spend working on homework problems will be strongly correlated with your success in the module.

Just like learning a natural language such as Arabic, there is no shortcut to master the skills in this course. To successfully complete this course will require a focused 160 hours that will involve dedication, concentration, sweat, frustration, possible tears and effort. All of these requirements and symptoms are signs of learning which involves per definition to be out of your comfort zone to allow you to grow so that you become comfortable with the new knowledge and skills to which this course will expose you i.e. extending your comfort zone to include what MKM321 entails.

6 Rules of assessment

Also see the examination regulations in the Year Books of the Faculty of Engineering, Built Environment and Information Technology (Part 1: Engineering).

Pass requirements: To pass this module a learner must

1. obtain a final mark of at least 50%,
2. must achieve a subminimum of 40% for the exam.

To obtain admission to the final examination, students must have submitted all weekly assignments, which will be assessed on whether students have engaged sufficiently with the assignments.

Any assignments that are graded as insufficient must be updated and resubmitted until a sufficient grade is awarded. The aim of the assignments are not to reward the correctness of the answer but to reward engagement and reflection on the work completed, i.e. reflection, interpretation and critical thinking is essential when completing these assignments. Students are encouraged to "play" with the assignments and communicate their findings. Please note the exam can only contribute a maximum of 20% to the final mark.

Calculation of the final mark: The final mark is based on continuous assessments. As unforeseen circumstances may arise during the semester given the current situation with Covid and load shedding that I cannot necessary anticipatet, the number of assignements and their respective weightings may change over the semester. Class test scope for MKM321 will always be up to what what we have covered in class up to that Monday morning, usually with a focus of the assignment that is due for that Monday. The current number of assignments and weightings are as follows:

- Exam (90 minutes): 20.0%
- Semester test 1 (90 minutes): 10%
- Semester test 2 (90 minutes): 10%
- Class Test 1 (Up to Background): 5.0%
- Class Test 2 (Up to Statically indeterminate structures): 5.0%
- Class Test 3 (Up to Displacement field, displacement boundary conditions, strain, stress fieds): 5.0%
- Class Test 4 (Up to Stress equilibrium, Strong form of the Solid Mechanics Problem, traction boundary conditions and traction reactions): 5.0%
- Class Test 5 (Up to Function approximation weighted residual method): 5.0%
- Class Test 6 (Up to Weak form of the Solid Mechanics Problem and approximate solutions): 5.0%
- Class Test 7 (Up to Numerical integration (Gauss quadrature) and single element finite element solutions original domain: 5.0%
- Class Test 8 (Up to 1D extension to 2D/3D finite elements, strain energy, deformation modes, work, verification, validation, Saint's Principal, consistent nodal loads, statically equivalent loads, full and reduced integration, boundary conditions, communicable modes and stress concentrations): 5.0%
- Class Test 9 (Up to 1D extension to 2D/3D finite elements, strain energy, deformation modes and work, verification, validation, Saint's Principal, consistent nodal loads, statically equivalent loads, full and reduced integration, boundary conditions, communicable modes and stress concentrations): 5.0%
- Class Test 10 (Up to Superposition, symmetric and anti-symmetric boundary conditions, volumetric and shear locking): 5.0%

- Class Test 11 (Up to Superposition, symmetric and anti-symmetric boundary conditions, volumetric and shear locking): 5.0%
- Class Test 12 (Up to Everything of the End of Week 13): 5.0%

Weekly class tests will count in total 60% towards your final grade. Due to load shedding, these tutorial tests will be made avaialble directly after the Monday tutorial session and are due within 48 hours, where you will supply the answers under a created Test on ClickUP. These tests will be automatically graded.

Weekly tutorial assignments are to be submitted but will only be graded to assess the level of engagement of the assignment which will be graded as sufficient of insufficient. To pass the course all assignments must be complered as sufficient. Any tutorials that are graded as insufficient must be expanded on until a sufficient grade is awarded. Note, the weekly tutorial assignments will run concurrently with the material that is covered in class in the week that it is released (usually Friday evening/Saturday morning depending on load shedding) as well as the following week. One tutorial hour will be spend to cover the tutorial assignment that will be due the next Monday morning at 8h00. The remaining tutorial hour will be used to cover the tutorial assignment that was submitted that morning, which is also the material the tutorial test will cover.

Examination entrance:

To obtain admission to the final examination, students must have submitted all weekly assignments, which will be assessed on whether students have engaged sufficiently with the assignments. Any assignments that are graded as insufficient must be updated and resubmitted until a sufficient grade is awarded. The aim of the assignments are not to reward the correctness of the answer but to reward engagement and reflection on the work completed, i.e. reflection, interpretation and critical thinking is essential when completing these assignments.

7 General

7.1 Submission of assignments

Your Jupyter-notebooks for the programming assignments and continuous assessment tests must be submitted electronically through ClickUP.

7.2 Module changes

While every effort has been made to ensure that this manual is correct, it is unavoidable that changes to both the organization and content of this module might be required from time to time. Hence this manual is **subject to change** by means of announcements on ClickUP.

7.3 Web site

Course material and assignments will be made available on ClickUP. It is the responsibility of the student to check the course webpage daily for announcements.

8 Module yearbook description

`Introduction to continuum mechanics. Kinematics of deformation and the strain tensor. Lagrangian and Eulerian descriptions. The stress tensor and equilibrium equations. Hooke's law for isotropic media. Strong form of Boundary Value Problem (BVP) of solid mechanics. Weak form of BVP of solid mechanics. Derivation of finite element equations using weighted residuals. Development of 2D elements.`

9 Module structure

Study theme and study units	*Notional hours*	*Contact sessions*
Articulation and extension on established knowledge:		
Differentiating statically determinate from indeterminate problems	12	3
Solving basic statically determinate from indeterminate problems using Python that will Develop Numerical Linear Algebra Python (module scipy) and Symbolic Python (module sympy)	12	3
Mathematical language and continuum models:		
traction vector, stress components, stress tensor, statically indeterminate problems, traction vector, stress components, stress tensor, strong form, closed form solutions, approximating functions at discrete points, approximating functions integral approach, weighted residual, weak form, 1D solutions arbitrary functions, 1D solutions structural problems	53	12
Regression by the finite element method:		
1-D Isoparametric formulation, numerical integration, assembling multiple elements, loadings, post-processing (strain,stress,reaction forces), interpreting results, Saint Venant's principle, linear superposition	53	15
Numerical literacy:		
Conceptual extension to 2-D elements using knowledge from 1D elements	6	2
2D: Saint Venant's Principle, Linear Superposition, Rigid Body Modes	6	1
2D: Plane Stress, Plane Strain, Symmetry, Anti-Symmetry	6	1
2D: Stress Concentrations and Infinite Stresses, V&V	6	1
2D: Structural Dynamics, Time Integration, Symmetry & Anti-Symmetry	6	1
TOTAL	**160**	**36**

9.1 Introduction, continuum mechanics models and mathematical language

9.1.1 Overview

Topic	*Source*	*Remarks*
Force and Moment Equilibrium	Class Notes & MOM 4	
Differential Equations of the Deflection Curve	Class Derivation & MOM 9.2	
Deflections by Integration	Class Notes & MOM 9.3 - 9.4	
Statically Indeterminate Problems	Class Notes & MOM 10.2, 10.4	
Introduction to Stress Equilibrium	C&B 1.1	Self study
Historical background	C&B 1.2	Self study
Traction vector	Class notes & MOM 2.6	
Stresses and equilibrium	C&B 1.4 & Class notes	
Boundary conditions	C&B 1.5	
Strain-displacement relations	C&B 1.6 & Class notes	
Stress-strain relations	C&B 1.7 & Class notes & MOM 7	
Mixed BVP of solid mechanics	Class notes	
Method of weighted residuals & Galerkin's method	C&B 1.10, Class notes	
Saint Venant's principle	C&B 1.11	
Von Mises stress	C&B 1.12	
Principle of superposition	C&B 1.13	
Matrix algebra	C&B 2.1, 2.2 & Note books	Self study
Python	Jupyter-Notebooks & MP213 Notes	Self study

MOM - Mechanics of Materials by Goodno and Gere (Prescribed Textbook of Prerequisite Course).

9.1.2 Key terms

Statically Indeterminate, linear superposition, traction vector, stress tensor, strain tensor, equilibrium conditions, symmetry, strain-displacement relations, plane stress, plane strain, mixed boundary value problem, strong form, weak form, Gauss's divergence theorem, method of weighted residuals, Gauss quadrature, Stress, strain, equilibrium, boundary conditions (prescribed tractions and prescribed displacements), strain-displacement relation, stress-strain relation, Saint Venant's principle, stress transformations, maximum shear, maximum tension, maximum compression, von Mises stress, matrix algebra, solving linear systems.

9.1.3 Mathematical background required

Integration by parts, divergence (Green-Gauss). Knowledge of vectors, matrices and scalars.

9.1.4 Study outcomes

You should have a basic understanding of statically indeterminate problems in structural mechanics and be able to solve basic statically indeterminate problems using linear superposition using the beam deflection tables using force and moment equilibrium. Being comfortable, with the examples covered in class and the tutorial assignments is sufficient. You will need to have a fundamental understanding of matrix algebra and be able to solve linear systems using sympy or numpy in Python.

You must be comfortable with displacement fields in one, two and three dimensions that describe the displacements of a continuous body as a field over the geometry of the undeformed body. You must be able to express internal force vectors on surfaces as traction vectors for surfaces defined by unit outward normals. You must be able to express internal force vectors on surfaces defined by unit outward normals in terms of stress components. You should be able to conduct basic stress transformations and calculate von Mises stresses.

Given any displacement field, material properties and problem geometry (1D, 2D Plane Stress, 2D Plane Strain and 3D), you must be able to compute the strains, and then stresses. Check if the stresses satisfy the equilibrium conditions inside the body, and the traction boundary conditions on the surface. You should be comfortable solving problems symbolically in Python using `sympy`, problems covered in class and the tutorial assignments is of sufficient complexity for the purposes of this course.

You should be able to pose the strong form of the mixed boundary value problem (BVP). This includes listing of all the unknowns to be solved, the equations available, and the boundary conditions that have to be satisfied. You must also be able to pose the weak form of the mixed BVP, including the restrictions on the weighting function.

You will have to understand the implications of Saint Venant's principle.

9.2 Solving Linear Systems Using Python

9.2.1 Overview

Topic	*Source*	*Remarks*
Matrix algebra	C&B 2.1	
Matrix algebra in Python	Jupyter-Notebooks	
Solving linear systems using Python	Jupyter-Notebooks	

9.2.2 Key terms

Matrix algebra. Hence all of the concepts and mathematical operations on vectors, matrices and scalars treated in C&B Chapter 2, e.g. positive definiteness, eigenvalues and eigenvectors, Gaussian elimination, symmetry, banded matrices, etc.

9.2.3 Study outcomes

You should be comfortable with the mathematical operators in this chapter, since these mathematical concept are fundamental to the remainder of the course. In addition, you should be comfortable using Python to work with vectors and matrices (numpy.arrays, indexing, slicing, etc.).

9.3 1-D problems

9.3.1 Overview

Topic	*Source*	*Remarks*
Introduction	C&B 3.1	
Integration by parts	Notes	
Finite element modeling	C&B 3.2	
Coordinates and shape functions	C&B 3.3, 3.9	
Isoparametric Formulation	C&B 8	
Galerkin's method (weighted residual)	C&B 3.5	
Assembly of tensors	C&B 3.6	
Equations and boundary conditions	C&B 3.8	
Linear and quadratic elements	C&B 3.9	
Numerical integration	C&B 8.3	
Assembly (up to two quadratic elements)	C&B 8.3	
Estimate Expected FEM result	C&B 3, Class, Assignments	
Critically Interpret Approximate FEM result	C&B 3, Class, Assignments	
Ability Critically Interpret claimed FEM result for Stated Problem	C&B 3, Class, Assignments	
Work and strain energy	CB 1, Jupyter-Notebook	

9.3.2 Key terms and Study outcomes

Global Coordinates, local coordinates, isoparametric formulation (relate x to ξ), shape function derivation, condition shape function for rigid body motion, check validity of proposed shape functions, implications of assumed displacement field, Galerkin's method (method of weighted residuals), numerical integration (exact and inexact integration), element stiffness matrix, global stiffness matrix, point loads, surface tractions and body forces, consistent nodal loads, elimination approach, apply non-zero prescribed displacements, solve reaction forces. Implication of non-constant bar area, implication of non-constant material properties, implication of middle node not being in middle of element in global coordinates, be able to compute element stiffness matrices and element load vectors given these variations. Understand work by external loads and strain energy. Estimate expected FEM result, critically interpret approximate FEM result, ability critically interpret claimed FEM result for stated problem. Relate displacement, strain and stress fields. Able to assemble at least two quadratic elements given global coordinate numbering by hand or Python.

9.4 Extending 1D FEM to 2D and Beyond: Finite element modelling

9.4.1 Overview

Topic	*Source*	*Remarks*
Conceptual Extension 1D to 2D	Relate C&B 3 to C&B 8	
Consistent Nodal Loads	C&B 8.2 & Jupyter-Notebook	
Statically Equivalent Nodal Loads	C&B 1.11 & Jupyter-Notebook	
Python 2D FEM Program	Class Notes & Jupyter-Notebook	
2D FEM Program Input File	Class Notes	
2D FEM Program Output	Class Notes	
Implications Linear Elastic Assumptions	Jupyter-Notebook	
Linear superposition in modelling	C&B 1.13, 4.5 & Jupyter-Notebook	
Saint Venant's principle in modelling	C&B 1.11 & Jupyter-Notebook	
Complex Geometry (Stress Concentrations)	C&B 1.11 & Jupyter-Notebook	
Modelling of Applied Force and Moment Resultants in FEM	Jupyter-Notebook	
Symmetry	C&B 4.5 & Assignments	
Anti-symmetry	C&B 4.5 & Assignments	
Verification	NAFEMS V&V & Assignments	
Reading Python Figures	Concept Card	
Work and strain energy	CB 1, Jupyter-Notebook	
Deformation Modes, Full and Reduced Integration	CB 3, Jupyter-Notebook	
Communicable Modes	Jupyter-Notebook	
Element Shear and Volumetric Locking	Jupyter-Notebook	
Validation and Singularities	NAFEMS V&V, Juputer-Notebook	

9.4.2 Key terms and Study outcomes

Extend 1D FEM knowledge conceptually to 2D FEM, which includes ideas such as unknown nodal displacements in 1D and 2D, shape function construction by roots, local nodal coordinates (-1 to 1 domain) in 1D and 2D, Numerical Integration in 1D and 2D (Exact and Inexact), Point Loads, Surface Tractions and Body Forces in 1D and 2D. Consistent nodal loads in 1D and 2D, Statically Equivalent Loads in 1D and 2D, Nodal Force and Moment Equilibrium in 1D and 2D. Fem Modelling: Implications of Large Deformations and Large Rotations in Linear Elastic FEM, Linear Superposition, Limits on Computed Stress Values, Saint Venant's principle. Estimate Expected Answer Before Conducting FEM, Interpret FEM Results, Compare Estimated Solution

Against FEM Solution (Verification) and making sure you properly read Figures Python Generate (see concept card). Understand that for complex geometries, mesh refinement influences both geometrical detail as well as more displacement field detail. Understand implications of stress concentrations, singularities (infinite stresses) in linear elastic models. Modelling of force and moment resultants in FEM, Identify and model symmetrical load cases, anti-symmetrical load cases, and symmetrical structures and BCs with no symmetry in loading using linear superposition. Understand and be able to compute the work done conducted by forces on structures, understand and be able to compute strain energy stored in structures. Interpreting the eigenvalues and vectors of stiffness matrices. Identify the number of rigid body modes of a stiffness matrix and relate it to full (three rigid body modes in 2D) or reduced (more than three rigid body modes in 2D) integration. Understand conceptual difference between full and reduced integration and the implications of communicable modes. Understand and identify limitations of some elements (four-noded quadrilateral) that includes shear and volumetric locking. Understand under which loadings and conditions shear and volumetric locking occur.

