Canvas LMS from Instructure is a cloud-based learning management system (LMS)
- It is a course management system that supports online learning and teaching
- It is an open-sourced cloud-based application designed to empower both teachers and students by making an engaging learning environment available to them.
- Canvas tools enhance teaching power and the student's learning experience, saving more time and effort and allowing greater focus on other priorities
- Canvas LMS offers easy-to-use and customizable teaching and learning experience.
- Canvas LMS is an excellent tool to organize learning. It is based on three main components, including a Dashboard, a high-level overview of top courses, and a Global navigation menu that provides access to the main features of the LMS.
- Canvas Common is another main highlight of the platform. It is a learning object repository where educators can supplement their classes with quizzes, modules, and courses. They can create discussions and documents in the repository. This feature provides educators with a fast track to course creation. Commons can also be used as a platform to share content within an institution.
- The best highlight of Canvas LMS is its Modules feature. This feature helps organize course content into units. Educators can stipulate prerequisites for modules in sequence. Students don’t see a course or unit until they unlock their prerequisites.
- Quiz Feature helps educators to take a quiz or assign an exam to any student if they require it at any place. Quizzes can be graded or ungraded, time-constraints, or self-paced, with long descriptive answers or multiple choices with true or false, etc.
- Gradebook is a feature that helps educators and administrators to grade students based on their assignments. The Grades can be imported or exported manually as well as automatically in a CSV file through Student Information System integration.
  - I created a course on Data Structures in semester 1
  - I have added all 60+ students to the course
  - In each course modules are created for unit wise. In each module I have added notes, ppts, Videos of online classes, other reference videos.
  - I conducted quiz and graded the students.
  - I given assignments and assignments are graded to the point 10.
Home Page of DataStructure Course in Canvas

In Introductory module I included syllabus, timetable, lecture plan, CO-PO etc

Modules in the course
### List of Students added to the course

<table>
<thead>
<tr>
<th>Name</th>
<th>Login ID</th>
<th>SIS ID</th>
<th>Section</th>
<th>Role</th>
<th>Last Activity</th>
<th>Total Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAI Pavani Kumar</td>
<td><a href="mailto:pavani200225@gmail.com">pavani200225@gmail.com</a></td>
<td></td>
<td>Data Structures</td>
<td>Student</td>
<td>Sep 7, 2020 at 7:44am</td>
<td>27:46</td>
</tr>
<tr>
<td>SAI Babu Anitha (She/Her)</td>
<td><a href="mailto:babu.anitha@gmail.com">babu.anitha@gmail.com</a></td>
<td></td>
<td>Data Structures</td>
<td>Student</td>
<td>Jan 4, 2021 at 7:04am</td>
<td>01:55:28</td>
</tr>
<tr>
<td>SAI Kalpana Bhat (He/Him)</td>
<td><a href="mailto:kalpana.bhat@gmail.com">kalpana.bhat@gmail.com</a></td>
<td></td>
<td>Data Structures</td>
<td>Student</td>
<td>Jan 4, 2021 at 7:28am</td>
<td>01:59:36</td>
</tr>
<tr>
<td>SAI Srinivasa (She/Her)</td>
<td><a href="mailto:srinivasa.srinivasa@gmail.com">srinivasa.srinivasa@gmail.com</a></td>
<td></td>
<td>Data Structures</td>
<td>Student</td>
<td>Feb 4, 2021 at 6:02pm</td>
<td>03:51:40</td>
</tr>
<tr>
<td>SAI Sri</td>
<td><a href="mailto:srinivasa.sri@gmail.com">srinivasa.sri@gmail.com</a></td>
<td></td>
<td>Data Structures</td>
<td>Student</td>
<td>Dec 12, 2020 at 8:00pm</td>
<td>38:50</td>
</tr>
<tr>
<td>SAI Gouthami</td>
<td><a href="mailto:gouthami.gouthami@gmail.com">gouthami.gouthami@gmail.com</a></td>
<td></td>
<td>Data Structures</td>
<td>Student</td>
<td>Aug 7, 2020 at 8:00pm</td>
<td>39:29:26</td>
</tr>
<tr>
<td>SAI T. Manasa (She/Her)</td>
<td><a href="mailto:tmanasa.tmanasa@gmail.com">tmanasa.tmanasa@gmail.com</a></td>
<td></td>
<td>Data Structures</td>
<td>Student</td>
<td>Sep 23, 2020 at 10:54am</td>
<td>01:49:36</td>
</tr>
<tr>
<td>SAI Sathya (She/Her)</td>
<td><a href="mailto:sathya.sathya@gmail.com">sathya.sathya@gmail.com</a></td>
<td></td>
<td>Data Structures</td>
<td>Student</td>
<td>Sep 23, 2020 at 10:54am</td>
<td>01:49:36</td>
</tr>
<tr>
<td>SAI Sathya (She/Her)</td>
<td><a href="mailto:sathya.sathya@gmail.com">sathya.sathya@gmail.com</a></td>
<td></td>
<td>Data Structures</td>
<td>Student</td>
<td>Sep 23, 2020 at 10:54am</td>
<td>01:49:36</td>
</tr>
</tbody>
</table>

### Attendance of the Students

#### Roll Call

- **Mark ALL Present**: Marked for all students.
- **Unmark ALL**: Unmarked for all students.

- **566 Sridevi**: Data Structures
- **571 Ch. Vijay Prasad**: Data Structures
- **573 G. Shravan**: Data Structures
- **574 G. Naga Sai Nitish**: Data Structures
- **577 N. Akash**: Data Structures
- **577 N. Akash**: Data Structures
- **578 G. Charith Reddy**: Data Structures
Graphs

- Basic Terminology
- Representation of Graphs
- Graph Traversals
  - DFS and BFS
- Spanning Trees
- Prims Algorithm
- Kruskals Algorithm

Assignment Given to the Students

Assignment-5 Convert Infix to Postfix and Prefix expressions

Convert the Following Infix Expression to Post fix and Prefix expression using Stack

1. (A + B) * (C + D)
2. (A - B/(C) * (A/ K-L))
3. 1+4/2*1+2/3/2
4. 9*(12+4)/5/6
5. a*b/(c/d+ e-f)

Take the screenshots of the solutions, save in one file and upload

Points 10

Submit a file upload
Sample Assignment submitted by student

Assignments Graded
Quiz Conducted to the Students

Quiz Paper
Students Analysis of Quiz which can be downloaded as csv file

Live Lecture videos uploaded in canvas for students reference

Algorithm to convert Infix to Postfix

1. Read all the symbols one by one from left to right in the given Infix Expression.
2. If the reading symbol is operator, then directly print it to the result (output).
3. If the reading symbol is left parenthesis ‘(’, then Push it on to the Stack.
4. If the reading symbol is right parenthesis ‘)’, then Pop all the contents of stack until the operator is popped and print each popped symbol to the result.
5. If the reading symbol isOperator ‘+’, ‘-‘, ‘*‘, ‘/‘, ‘%‘, ‘^‘, ‘<<‘, ‘>>‘, then Push it on to the Stack.
6. However, first pop the operations which are already on the stack that have higher or equal precedence than current operator and print them to the result.

Activate Windows
Go to PC Settings to activate Windows.
Other videos for students reference
Outcome: Consistency of learning. Delivering course through an LMS keeps the content centralized so all the students have a single source of content, instructions and questions.