

COURSE CODE	COURSE TITLE	L	T	P	C
1152IT122	OPEN SOURCE COMPUTING	3	0	0	3

**Course Category:**

~~Foundation (0) / Program Core (1) / Program Elective (2) / Allied Elective (3) / University Elective (4) / Value Education Elective (5) / Independent Learning (6) / Industry Higher Learning Institute Interaction (7).~~

**a.Preamble**

Open Source Computing is to provide exposure in FOSS and to develop open source software for society.

**b.Prerequisite Courses:**

Unix Programming.

**c.Related Courses:**

Web technology

**d.Course Educational Objectives :**

The student should be made to:

- Be exposed to the context and operation of free and open source software (FOSS) communities and associated software projects.
- Be familiar with participating in a FOSS project
- Learn scripting language like Python or Perl
- Learn programming language like Ruby
- Learn some important FOSS tools and techniques

**e.Course Outcomes :**

Upon the successful completion of the course, students will be able to:

CO Nos.	Course Outcomes	Knowledge Level (Based on revised Bloom's Taxonomy)
CO1	Install and run open-source operating systems.	K2
CO2	Gather information about Free and Open Source Software projects from software releases and from sites on the internet.	K2
CO3	modify one or more Free and Open Source Software projects.	K1
CO4	Contribute software to and interact with Free and Open Source Software development projects.	K2

**UNIT I PHILOSOPHY**

9

Four degrees of freedom - FOSS Licensing Models - FOSS Licenses – GPL- AGPL- LGPL - FDL - Implications – FOSS examples-Open Source, Free Software, Free Software vs. Open Source software, Public Domain Software- History : BSD, The Free Software Foundation and the GNU Project.

<b>UNIT II</b>	<b>LINUX</b>	<b>9</b>
Linux Installation and Hardware Configuration – Boot Process-The Linux Loader (LILO) - The Grand Unified Bootloader (GRUB) - Dual-Booting Linux and other Operating System - Boot-Time Kernel Options- X Windows System Configuration-System Administration – Backup and Restore Procedures- Strategies for keeping a Secure Server.		
<b>UNIT III</b>	<b>PROGRAMMING LANGUAGES</b>	<b>9</b>
Programming using languages like Python , Perl , Ruby		
<b>UNIT IV</b>	<b>PROGRAMMING TOOLS AND TECHNIQUES</b>	<b>9</b>
Usage of design Tools like Argo UML or equivalent, Version Control Systems like Git or equivalent–Bug Tracking Systems- Package Management Systems		
<b>UNIT V</b>	<b>FOSS CASE STUDIES</b>	<b>9</b>
Open Source Software Development - Case Study – Libreoffice –Samba-Mozilla (Firefox), Wikipedia, Joomla, GCC, Open Office.		
		<b>TOTAL: 45 PERIODS</b>

## **Learning Resources**

### **i. TEXT BOOKS**

1. Ellen Siever, Stephen Figgins, Robert Love, Arnold Robbins, “Linux in a Nutshell”, Sixth Edition, OReilly Media, 2009.

### **ii. REFERENCES:**

1. Philosophy of GNU URL: <http://www.gnu.org/philosophy/>.
2. Linux Administration URL: <http://www.tldp.org/LDP/lame/LAME/linux-admin-made-easy/>.
3. The Python Tutorial available at <http://docs.python.org/2/tutorial/>.
4. Perl Programming book at <http://www.perl.org/books/beginning-perl/>.
5. Ruby programming book at <http://ruby-doc.com/docs/ProgrammingRuby/>.
6. Version control system URL: <http://git-scm.com/>.
7. Samba: URL : <http://www.samba.org/>.
8. Libre office: <http://www.libreoffice.org/>.