

A Free Software course for undergraduates

An experience report

Ralf Treinen

UFR Informatique
Université Paris Diderot
treinen@pps.jussieu.fr



The course

- ▶ Course given once per year since 2007
- ▶ Format: 2h per week in a classroom.
- ▶ Audience:
 - ▶ Second-year undergraduate students at Université Paris-7
 - ▶ Students from all subject areas, except ...
 - ▶ ... Computer Science (they have their own course in the 4th year)
- ▶ This is one of the courses that students may take outside of their specific curriculum.
- ▶ Previous years: 6 to 20 students per year.

The course

- ▶ Course given once per year since 2007
- ▶ Format: 2h per week in a classroom.
- ▶ Audience:
 - ▶ Second-year undergraduate students at Université Paris-7
 - ▶ Students from all subject areas, except ...
 - ▶ ... Computer Science (they have their own course in the 4th year)
- ▶ This is one of the courses that students may take outside of their specific curriculum.
- ▶ Previous years: 6 to 20 students per year.

The course

- ▶ Course given once per year since 2007
- ▶ Format: 2h per week in a classroom.
- ▶ Audience:
 - ▶ Second-year undergraduate students at Université Paris-7
 - ▶ Students from all subject areas, except ...
 - ▶ ... Computer Science (they have their own course in the 4th year)
- ▶ This is one of the courses that students may take outside of their specific curriculum.
- ▶ Previous years: 6 to 20 students per year.

The course

- ▶ Course given once per year since 2007
- ▶ Format: 2h per week in a classroom.
- ▶ Audience:
 - ▶ Second-year undergraduate students at Université Paris-7
 - ▶ Students from all subject areas, except ...
 - ▶ ... Computer Science (they have their own course in the 4th year)
- ▶ This is one of the courses that students may take outside of their specific curriculum.
- ▶ Previous years: 6 to 20 students per year.

The course

- ▶ Course given once per year since 2007
- ▶ Format: 2h per week in a classroom.
- ▶ Audience:
 - ▶ Second-year undergraduate students at Université Paris-7
 - ▶ Students from all subject areas, except ...
 - ▶ ... Computer Science (they have their own course in the 4th year)
- ▶ This is one of the courses that students may take outside of their specific curriculum.
- ▶ Previous years: 6 to 20 students per year.

The course

- ▶ Course given once per year since 2007
- ▶ Format: 2h per week in a classroom.
- ▶ Audience:
 - ▶ Second-year undergraduate students at Université Paris-7
 - ▶ Students from all subject areas, except ...
 - ▶ ... Computer Science (they have their own course in the 4th year)
- ▶ This is one of the courses that students may take outside of their specific curriculum.
- ▶ Previous years: 6 to 20 students per year.

The course

- ▶ Course given once per year since 2007
- ▶ Format: 2h per week in a classroom.
- ▶ Audience:
 - ▶ Second-year undergraduate students at Université Paris-7
 - ▶ Students from all subject areas, except ...
 - ▶ ... Computer Science (they have their own course in the 4th year)
- ▶ This is one of the courses that students may take outside of their specific curriculum.
- ▶ Previous years: 6 to 20 students per year.

Previous knowledge of students

- ▶ Everybody uses a computer (in particular for word processing)
- ▶ Very few have a completely free software installation (GNU/Linux) on their own computer.
- ▶ University computers used for teaching are FreeBSD, but students are rarely aware of the fact that this is free.
- ▶ At home, students often use isolated free software (OpenOffice)
- ▶ Wikipedia (as another free project) is more “visible” with its particular philosophy

Previous knowledge of students

- ▶ Everybody uses a computer (in particular for word processing)
- ▶ Very few have a completely free software installation (GNU/Linux) on their own computer.
- ▶ University computers used for teaching are FreeBSD, but students are rarely aware of the fact that this is free.
- ▶ At home, students often use isolated free software (OpenOffice)
- ▶ Wikipedia (as another free project) is more “visible” with its particular philosophy

Previous knowledge of students

- ▶ Everybody uses a computer (in particular for word processing)
- ▶ Very few have a completely free software installation (GNU/Linux) on their own computer.
- ▶ University computers used for teaching are FreeBSD, but students are rarely aware of the fact that this is free.
- ▶ At home, students often use isolated free software (OpenOffice)
- ▶ Wikipedia (as another free project) is more “visible” with its particular philosophy

Previous knowledge of students

- ▶ Everybody uses a computer (in particular for word processing)
- ▶ Very few have a completely free software installation (GNU/Linux) on their own computer.
- ▶ University computers used for teaching are FreeBSD, but students are rarely aware of the fact that this is free.
- ▶ At home, students often use isolated free software (OpenOffice)
- ▶ Wikipedia (as another free project) is more “visible” with its particular philosophy

Previous knowledge of students

- ▶ Everybody uses a computer (in particular for word processing)
- ▶ Very few have a completely free software installation (GNU/Linux) on their own computer.
- ▶ University computers used for teaching are FreeBSD, but students are rarely aware of the fact that this is free.
- ▶ At home, students often use isolated free software (OpenOffice)
- ▶ Wikipedia (as another free project) is more “visible” with its particular philosophy

First part of the course: lecture (8 weeks)

- ▶ **Introduction**
- ▶ Foundation of computing (3 lectures) : programming (Logo), issues in software engineering, operating systems, ...
- ▶ History of free software
- ▶ Philosophy of free and open source software, legal aspects (licences)
- ▶ Software development in the FOSS world - the cathedral and the bazaar
- ▶ Debian as an example of a volunteer project
- ▶ Economics of free software:
 - ▶ How is it possible to make money by publishing free software?
 - ▶ Why is it important for a client to buy a software solution that is based on FOSS?

First part of the course: lecture (8 weeks)

- ▶ Introduction
- ▶ Foundation of computing (3 lectures) : programming (Logo), issues in software engineering, operating systems, ...
- ▶ History of free software
- ▶ Philosophy of free and open source software, legal aspects (licences)
- ▶ Software development in the FOSS world - the cathedral and the bazaar
- ▶ Debian as an example of a volunteer project
- ▶ Economics of free software:
 - ▶ How is it possible to make money by publishing free software?
 - ▶ Why is it important for a client to buy a software solution that is based on FOSS?

First part of the course: lecture (8 weeks)

- ▶ Introduction
- ▶ Foundation of computing (3 lectures) : programming (Logo), issues in software engineering, operating systems, ...
- ▶ History of free software
- ▶ Philosophy of free and open source software, legal aspects (licences)
- ▶ Software development in the FOSS world - the cathedral and the bazaar
- ▶ Debian as an example of a volunteer project
- ▶ Economics of free software:
 - ▶ How is it possible to make money by publishing free software?
 - ▶ Why is it important for a client to buy a software solution that is based on FOSS?

First part of the course: lecture (8 weeks)

- ▶ Introduction
- ▶ Foundation of computing (3 lectures) : programming (Logo), issues in software engineering, operating systems, ...
- ▶ History of free software
- ▶ Philosophy of free and open source software, legal aspects (licences)
- ▶ Software development in the FOSS world - the cathedral and the bazaar
- ▶ Debian as an example of a volunteer project
- ▶ Economics of free software:
 - ▶ How is it possible to make money by publishing free software?
 - ▶ Why is it important for a client to buy a software solution that is based on FOSS?

First part of the course: lecture (8 weeks)

- ▶ Introduction
- ▶ Foundation of computing (3 lectures) : programming (Logo), issues in software engineering, operating systems, ...
- ▶ History of free software
- ▶ Philosophy of free and open source software, legal aspects (licences)
- ▶ Software development in the FOSS world - the cathedral and the bazaar
- ▶ Debian as an example of a volunteer project
- ▶ Economics of free software:
 - ▶ How is it possible to make money by publishing free software?
 - ▶ Why is it important for a client to buy a software solution that is based on FOSS?

First part of the course: lecture (8 weeks)

- ▶ Introduction
- ▶ Foundation of computing (3 lectures) : programming (Logo), issues in software engineering, operating systems, ...
- ▶ History of free software
- ▶ Philosophy of free and open source software, legal aspects (licences)
- ▶ Software development in the FOSS world - the cathedral and the bazaar
- ▶ Debian as an example of a volunteer project
- ▶ Economics of free software:
 - ▶ How is it possible to make money by publishing free software?
 - ▶ Why is it important for a client to buy a software solution that is based on FOSS?

First part of the course: lecture (8 weeks)

- ▶ Introduction
- ▶ Foundation of computing (3 lectures) : programming (Logo), issues in software engineering, operating systems, ...
- ▶ History of free software
- ▶ Philosophy of free and open source software, legal aspects (licences)
- ▶ Software development in the FOSS world - the cathedral and the bazaar
- ▶ Debian as an example of a volunteer project
- ▶ Economics of free software:
 - ▶ How is it possible to make money by publishing free software?
 - ▶ Why is it important for a client to buy a software solution that is based on FOSS?

First part of the course: lecture (8 weeks)

- ▶ Introduction
- ▶ Foundation of computing (3 lectures) : programming (Logo), issues in software engineering, operating systems, ...
- ▶ History of free software
- ▶ Philosophy of free and open source software, legal aspects (licences)
- ▶ Software development in the FOSS world - the cathedral and the bazaar
- ▶ Debian as an example of a volunteer project
- ▶ Economics of free software:
 - ▶ How is it possible to make money by publishing free software?
 - ▶ Why is it important for a client to buy a software solution that is based on FOSS?

First part of the course: lecture (8 weeks)

- ▶ Introduction
- ▶ Foundation of computing (3 lectures) : programming (Logo), issues in software engineering, operating systems, ...
- ▶ History of free software
- ▶ Philosophy of free and open source software, legal aspects (licences)
- ▶ Software development in the FOSS world - the cathedral and the bazaar
- ▶ Debian as an example of a volunteer project
- ▶ Economics of free software:
 - ▶ How is it possible to make money by publishing free software?
 - ▶ Why is it important for a client to buy a software solution that is based on FOSS?

First part of the course: lecture (8 weeks)

- ▶ Introduction
- ▶ Foundation of computing (3 lectures) : programming (Logo), issues in software engineering, operating systems, ...
- ▶ History of free software
- ▶ Philosophy of free and open source software, legal aspects (licences)
- ▶ Software development in the FOSS world - the cathedral and the bazaar
- ▶ Debian as an example of a volunteer project
- ▶ Economics of free software:
 - ▶ How is it possible to make money by publishing free software?
 - ▶ Why is it important for a client to buy a software solution that is based on FOSS?

Second part of the course: student projects (4-5 weeks)

- ▶ Each student has to a small research project on a FOSS topic.
- ▶ Presentation in class (about 30 minutes), plus questions from the teacher and the class.
- ▶ The teacher suggests a list of subjects, students are invited to propose their own subject (done by about 50% of the students)
- ▶ Ideally, each students meets twice with the teacher before the presentation to discuss literature, and structure of the talk.
- ▶ Students are encouraged to be active (try to install a software, contribute something to a free project like Wikipedia or openstreetmap)

Second part of the course: student projects (4-5 weeks)

- ▶ Each student has to a small research project on a FOSS topic.
- ▶ Presentation in class (about 30 minutes), plus questions from the teacher and the class.
- ▶ The teacher suggests a list of subjects, students are invited to propose their own subject (done by about 50% of the students)
- ▶ Ideally, each students meets twice with the teacher before the presentation to discuss literature, and structure of the talk.
- ▶ Students are encouraged to be active (try to install a software, contribute something to a free project like Wikipedia or openstreetmap)

Second part of the course: student projects (4-5 weeks)

- ▶ Each student has to a small research project on a FOSS topic.
- ▶ Presentation in class (about 30 minutes), plus questions from the teacher and the class.
- ▶ The teacher suggests a list of subjects, students are invited to propose their own subject (done by about 50% of the students)
- ▶ Ideally, each students meets twice with the teacher before the presentation to discuss literature, and structure of the talk.
- ▶ Students are encouraged to be active (try to install a software, contribute something to a free project like Wikipedia or openstreetmap)

Second part of the course: student projects (4-5 weeks)

- ▶ Each student has to a small research project on a FOSS topic.
- ▶ Presentation in class (about 30 minutes), plus questions from the teacher and the class.
- ▶ The teacher suggests a list of subjects, students are invited to propose their own subject (done by about 50% of the students)
- ▶ Ideally, each students meets twice with the teacher before the presentation to discuss literature, and structure of the talk.
- ▶ Students are encouraged to be active (try to install a software, contribute something to a free project like Wikipedia or openstreetmap)

Second part of the course: student projects (4-5 weeks)

- ▶ Each student has to a small research project on a FOSS topic.
- ▶ Presentation in class (about 30 minutes), plus questions from the teacher and the class.
- ▶ The teacher suggests a list of subjects, students are invited to propose their own subject (done by about 50% of the students)
- ▶ Ideally, each students meets twice with the teacher before the presentation to discuss literature, and structure of the talk.
- ▶ Students are encouraged to be active (try to install a software, contribute something to a free project like Wikipedia or openstreetmap)

Example projects

- ▶ **Richard Stallman and the FSF**
- ▶ LIMUX (Linux migration of the city of Munich)
- ▶ Presentation of particular free software projects (OpenOffice, Firefox, various games, various audio and video software).
Popular with students when they can talk about the software they use for their hobby
- ▶ Distributions : Redhat, Mandriva, Ubuntu, ...
- ▶ Software Patents
- ▶ Other free projects : Wikipedia, wikitravel, openstreetmap, ...
- ▶ Social aspects : Women in Free Software

Example projects

- ▶ Richard Stallman and the FSF
- ▶ LIMUX (Linux migration of the city of Munich)
- ▶ Presentation of particular free software projects (OpenOffice, Firefox, various games, various audio and video software).
Popular with students when they can talk about the software they use for their hobby
- ▶ Distributions : Redhat, Mandriva, Ubuntu, ...
- ▶ Software Patents
- ▶ Other free projects : Wikipedia, wikitravel, openstreetmap, ...
- ▶ Social aspects : Women in Free Software

Example projects

- ▶ Richard Stallman and the FSF
- ▶ LIMUX (Linux migration of the city of Munich)
- ▶ Presentation of particular free software projects (OpenOffice, Firefox, various games, various audio and video software).
Popular with students when they can talk about the software they use for their hobby
- ▶ Distributions : Redhat, Mandriva, Ubuntu, ...
- ▶ Software Patents
- ▶ Other free projects : Wikipedia, wikitravel, openstreetmap, ...
- ▶ Social aspects : Women in Free Software

Example projects

- ▶ Richard Stallman and the FSF
- ▶ LIMUX (Linux migration of the city of Munich)
- ▶ Presentation of particular free software projects (OpenOffice, Firefox, various games, various audio and video software).
Popular with students when they can talk about the software they use for their hobby
- ▶ Distributions : Redhat, Mandriva, Ubuntu, ...
- ▶ Software Patents
- ▶ Other free projects : Wikipedia, wikitravel, openstreetmap, ...
- ▶ Social aspects : Women in Free Software

Example projects

- ▶ Richard Stallman and the FSF
- ▶ LIMUX (Linux migration of the city of Munich)
- ▶ Presentation of particular free software projects (OpenOffice, Firefox, various games, various audio and video software).
Popular with students when they can talk about the software they use for their hobby
- ▶ Distributions : Redhat, Mandriva, Ubuntu, ...
- ▶ Software Patents
- ▶ Other free projects : Wikipedia, wikitravel, openstreetmap, ...
- ▶ Social aspects : Women in Free Software

Example projects

- ▶ Richard Stallman and the FSF
- ▶ LIMUX (Linux migration of the city of Munich)
- ▶ Presentation of particular free software projects (OpenOffice, Firefox, various games, various audio and video software).
Popular with students when they can talk about the software they use for their hobby
- ▶ Distributions : Redhat, Mandriva, Ubuntu, ...
- ▶ Software Patents
- ▶ Other free projects : Wikipedia, wikitravel, openstreetmap, ...
- ▶ Social aspects : Women in Free Software

Example projects

- ▶ Richard Stallman and the FSF
- ▶ LIMUX (Linux migration of the city of Munich)
- ▶ Presentation of particular free software projects (OpenOffice, Firefox, various games, various audio and video software).
Popular with students when they can talk about the software they use for their hobby
- ▶ Distributions : Redhat, Mandriva, Ubuntu, ...
- ▶ Software Patents
- ▶ Other free projects : Wikipedia, wikitravel, openstreetmap, ...
- ▶ Social aspects : Women in Free Software

Stocktaking

- ▶ This is a course that is demanding (a lot of interaction), but nice to give.
- ▶ The students are interested by the course (one needs to keep the right non-technical level).
- ▶ The most interesting part for everybody is the project phase (which provides a natural limit on the class size)

Stocktaking

- ▶ This is a course that is demanding (a lot of interaction), but nice to give.
- ▶ The students are interested by the course (one needs to keep the right non-technical level).
- ▶ The most interesting part for everybody is the project phase (which provides a natural limit on the class size)

Stocktaking

- ▶ This is a course that is demanding (a lot of interaction), but nice to give.
- ▶ The students are interested by the course (one needs to keep the right non-technical level).
- ▶ The most interesting part for everybody is the project phase (which provides a natural limit on the class size)