The image shows a login interface for 'OPUS college'. On the left is a dark blue square logo with the word 'OPUS' at the top, a stylized white sunburst icon in the middle, and the word 'college' at the bottom. To the right of the logo, the text 'Enter:' is positioned above two input fields. The first field is labeled 'user name:' and contains the text 'Nkonda'. The second field is labeled 'password:' and contains seven black dots. Below the password field is a blue 'Submit' button with a white mouse cursor icon pointing at it.

OPUS

college

Enter:

user name: Nkonda

password: ●●●●●●●

Submit

In this instruction we will show you how to register Courses (Subjects) in OPUS-College. This will involve the following steps:

1. Enter basic information (title, etc...) of the course
2. Enter the content description of the course
3. Enter the lecturers of the course
4. Linking the course to its Programme(s) of Study
5. Define the examination structure for the course.

A course can be registered by either the Dean or the Head of Department. As usual we login as Dean.

```
Version module: 3.3
Version module database: 3.18
```

useOfSubsidies: N

To enter a course, go to
"Curriculum - Courses -
Add"

[back to overview](#) > **new**
Course
Course
Second level unit

Office of the Dean (Parent/Child level 1) ▼

Primary study

-choose- ▼

Code

If empty, a code is generated

Name

Weight/Credit

0.0

The weight is used for

Study time

-choose- ▼

Max. no. of participants

0

Academic year

-choose- ▼

Result type

Default ▼

Active

yes ▼

Submit

You now get the screen to enter basic information on the course: the primary study the course belongs to, the name (title) of the course, its weight in a crediting system you may use, the study time (day, evening), the maximum number of participants (if it applies), the academic year the course belongs to and the result type for the course (if used by your institution).

Follow the instruction to fill in this screen.

[back to overview](#) > **new**
Course
Course
Second level unit

Department of Computer Sciences (Parent ▼

Primary study

Computer Science ▼

Code

S007032012153048

If empty, code is generated (which can be overruled)

Name

Programming 1

Weight/Credit

20

Study time

-choose- ▼

Max. no. of participants

0

Academic year

-choose- ▼

Result type

Default ▼

Active

yes ▼

Submit

The "Weight" of the course indicates the importance of the course in your crediting system (if you have one).

[back to overview](#) > new

Course

Course

Second level unit

Department of Computer Sciences (Parent ▼

Primary study

Computer Science ▼

Code

S007032012153048

If empty, a code is generated (which can be overruled)

Name

Programming 1

Weight/Credit

2.0

The valid weight/credit format is 999.9

Study time

Daytime ▼

Max. no. of participants

0

Academic year

2013 ▼

Result type

Default ▼

Active

yes ▼

Submit

[back to overview](#) > **Programming 1**
[Course](#) | [Content](#) | [Lecturers](#) | [Modes of delivery](#) | [Programmes of study](#) | [Examinations](#)

Course

Second level unit

Primary study

Code

Name

Weight/Credit

Study time

Max. no.

Academic year

Result type

Active

Computer Sciences (Parent)

After filling in the basic information, new tabs appear on top of the screen, indicating more sections of information on the course.

If empty, a code is generated (which can be overruled)

The valid weight/credit format is 999.9

2013

Default

yes

Submit

[back to overview](#) > **Programming 1**
[Course](#) | [Content](#) | [Lecturers](#) | [Modes of delivery](#) | [Programmes of study](#) | [Examinations](#)
Course

Second level

Primary studies

Department of Computer Sciences (Parent) ▼

Computer Science ▼

007032012153048

If empty, a code is generated (which can be overruled)

Programming 1

1.0

The valid weight/credit format is 999.9

Daytime ▼

2013 ▼

Default ▼

yes ▼

Submit

First tab concerns a description of the content of the course. Click on the tab to go to the corresponding screen.

Active

[back to overview](#) > Programming 1

Course

Content

Lecturers

Modes of delivery

Programmes of study

Examinations

Content

first press save-button in WYSIWYG widget, then press submit button

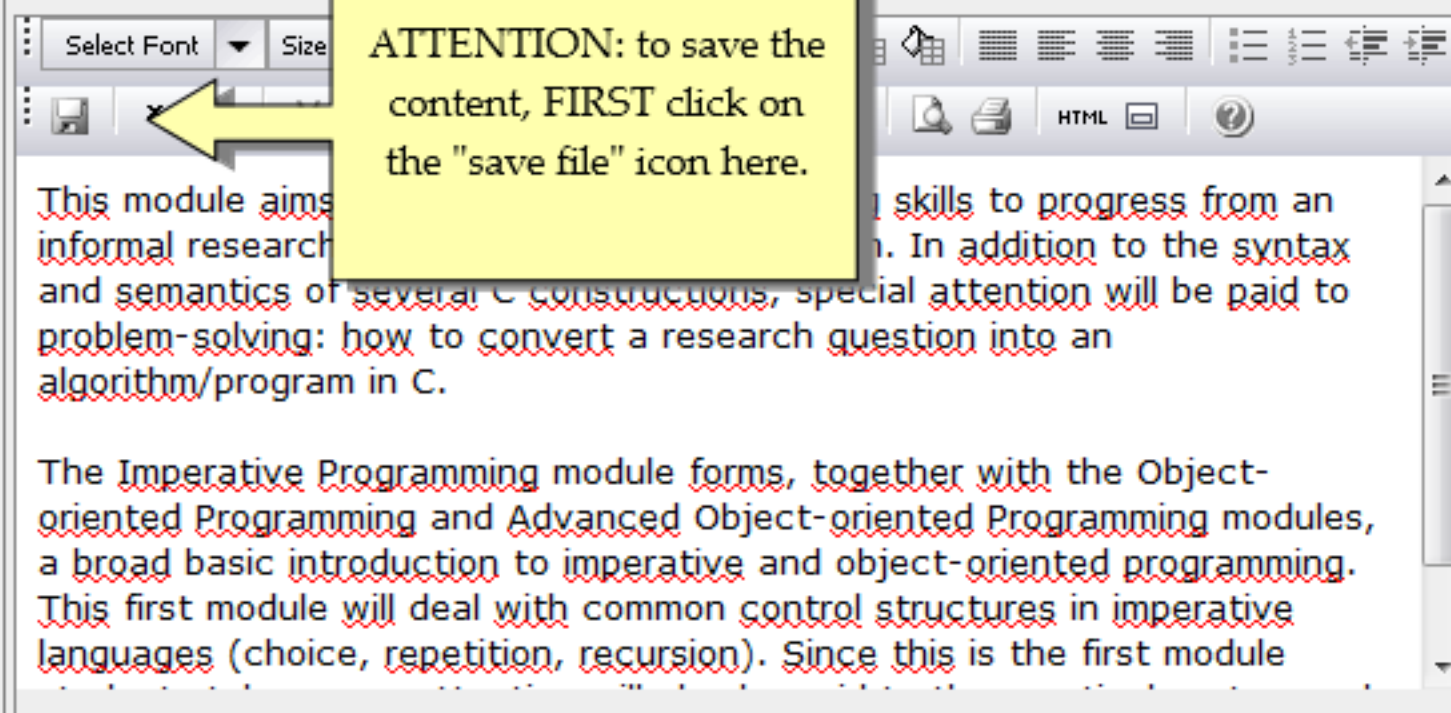
Submit

You now get an input field where you can either type the description of (the content of) the course or copy and paste the description, e.g. from a text file. This is what we have done in this example.

[back to overview](#) > **Programming 1**
[Course](#) | [Content](#) | [Lecturers](#) | [Modes of delivery](#) | [Programmes of study](#) | [Examinations](#)

Content

first press save-button in WYSIWYG widget, then press submit button



ATTENTION: to save the content, **FIRST** click on the "save file" icon here.

This module aims to develop skills to progress from an informal research. In addition to the syntax and semantics of several C constructions, special attention will be paid to problem-solving: how to convert a research question into an algorithm/program in C.

The Imperative Programming module forms, together with the Object-oriented Programming and Advanced Object-oriented Programming modules, a broad basic introduction to imperative and object-oriented programming. This first module will deal with common control structures in imperative languages (choice, repetition, recursion). Since this is the first module

And next click on the "Submit" button here.

[back to overview](#) > **Programming 1**[Course](#)[Content](#)[Lecturers](#)[Modes of delivery](#)[Programmes of study](#)[Examinations](#)**Lecturers for this course**[add](#)

There are no
lecturers registered yet.
You must register at least
one lecturer.

Next step is to register the
teachers of the course. Click
on this tab here.

[back to overview](#) > **Programming 1**[Course](#)[Content](#)[Lecturers](#)[Modes of delivery](#)[Programmes of study](#)[Examinations](#)**Lecturers for this course**

And then click on the
"add" button here.

There are no subject teachers chosen.
You must add at least one.

[back to overview](#) > [Programming 1](#) > **add/edit Courses taught****Lecturer for this course****Second level unit**

Department of Computer Sciences (Parent ▼)

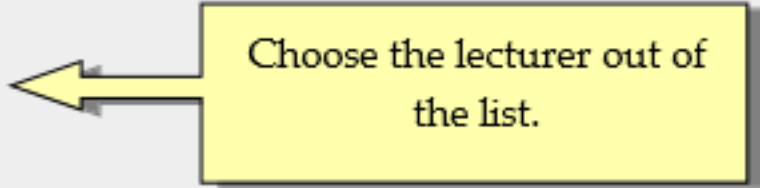
Lecturer

-choose-

-choose-

Jones, Mary

Katucha, Allan

Active Choose the lecturer out of
the list.

[back to overview](#) > **Programming 1**
[Course](#) | [Content](#) | [Lecturers](#) | [Modes of delivery](#) | [Programmes of study](#) | [Examinations](#)
Lecturers for this course
Description
Allan Katucha

Active

Y

[add](#)

✕

Next piece of information on the course is the "Mode(s) of delivery", i.e. the way the course is taught.

erlands & Ministry of Education and Culture, Mozambique

[back to overview](#) > **Programming 1**[Course](#)[Content](#)[Lecturers](#)[Modes of delivery](#)[Programmes of study](#)[Examinations](#)**Modes of delivery for this course**[add](#)

[back to overview](#) > [Programming 1](#) add/edit Mode of delivery

edit

Modes of delivery**Mode of delivery**

-choose-

Active

yes

Submit

Choose the mode out of
the list.

[back to overview](#) > [Programming 1](#) add/edit Mode of delivery

edit

Modes of delivery

Mode of delivery

Active

-choose-

-choose-

e-learning

experiment

group work

individual assistance by lecturer

lab/practical

literature

paper

project

self study

seminar

workshop

It is possible that a course has more than one "Mode of delivery".
As an example, this course also has a "lab/practical" part.

powered

ds & Ministry of Edu

[back to overview](#) > **Programming 1**
[Course](#) | [Content](#) | [Lecturers](#) | [Modes of delivery](#) | [Programmes of study](#) | [Examinations](#)
Modes of delivery for this course

Description

lab/practical

lecture

Active

Y

Y

add

✕

✕

Next thing to do is to link the course to its Programme(s) of Study. A course can be linked to more than one Programme of Study. E.g. the course Statistics may be common to more than one Programme.

[back to overview](#) > **Programming 1**[Course](#)[Content](#)[Lecturers](#)[Modes of delivery](#)[Programmes of study](#)[Examinations](#)**programmes of study for this course**[add](#)

[back to overview](#) > [Programming 1](#) > **add/edit Course / Programme of study**

edit

Course / Programme of study

Second level unit	Department of Computer Sciences (Parent ▼)
Course	Programming 1
Academic year	2013
Study	Computer Science ▼
Programme type	-choose- ▼
Category of study/Study time	-choose- ▼
Cardinal time unit number	Any ▼
Rigidity	-choose- ▼
Major / Minor	-choose- ▼
Active	yes ▼
	<input type="button" value="Submit"/>

By means of these fields you can choose the Programme of Study the course should be linked to (and in which time unit - semester, trimester).

Just follow the example.

[back to overview](#) > [Programming 1](#) > **add/edit Course / Programme of study**

edit

Course / Programme of study

Second level unit	Department of Computer Sciences (Parent ▼)
Course	Programming 1
Academic year	2013
Study	Computer Science ▼
Programme type	Bachelor of science ▼
Category of study/Study time	regular programme /Daytime ▼
Cardinal time unit number	1 ▼
Rigidity	compulsory ▼
Major / Minor	major ▼
Active	yes ▼
	<input type="button" value="Submit"/>

The "Major/Minor" distinction of Programmes of Study may not be implemented at your institution, in which case you will not see this field.

[back to overview](#) > [Programming 1](#) > add/edit Course / Programme of study[edit](#)**Course / Programme of study**

Second level unit	Department of Computer Sciences (Parent ▼)
Course	Programming 1
Academic year	2013
Study	Computer Science ▼
Programme type	Bachelor of science ▼
Category of study/Study time	regular programme /Daytime ▼
Cardinal time unit number	1 ▼
Rigidity	compulsory ▼
Major / Minor	major ▼
Active	yes ▼

[Submit](#)

[back to overview](#) > [Programming 1](#) > add/edit Course / Programme of study

edit

Course / Pro

Second level

Course

Academic year

Study

Programme type

Category of study/Study time

Cardinal time unit number

Rigidity

Major / Minor

Active

Prerequisite courses

Code

Description

Academic year

Active

Click on the name of the course to go back to its main menu.

Sciences (Parent ▼)

Bachelor of science ▼

regular programme /Daytime ▼

1 ▼

compulsory ▼

major ▼

yes ▼

Submit

add

[back to overview](#) > **Programming 1**
[Course](#) | [Content](#) | [Lecturers](#) | [Modes of delivery](#) | [Programmes of study](#) | [Examinations](#)

programmes of study for this course

Description

Computer
Science -
Bachelor of
science

Last, but really not least, thing to do is to define the Examination Structure of the course. This is quite complex and so requires a bit of introduction.

In OPUS-College it is possible to define a 3-level examination structure, like this:

COURSE (level 1 -- that may be subdivided in:)

|

|_ FINAL EXAM (level 2 -- that may e.g. be subdivided in:)

|_ Oral part (level 3)

|_ Written part (level 3)

|

|_ PERIODICAL TESTS (continuous assessment) (level 2, that may be subdivided in:)

|_ Test 1

|_ Test 2

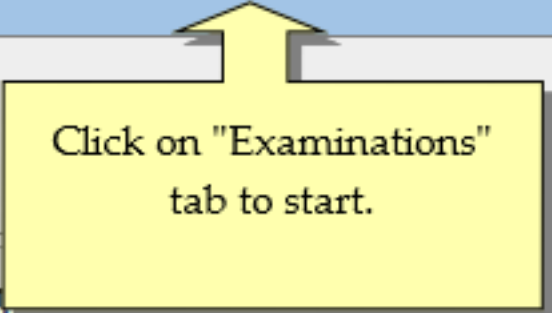
etc.....

You CAN but you DO NOT HAVE TO use all the levels. So it is perfectly possible to only and directly put results on the course level, or only use 2 levels, etc... All combinations are in principle possible.

In the next part of the instruction we will show the most complex structure: all 3 levels.

[back to overview](#) > **Programming 1**[Course](#)[Content](#)[Lecturers](#)[Modes of delivery](#)[Programmes of study](#)[Examinations](#)**Examinations for this course**

the total percentage is not 100 %

[add](#)

Click on "Examinations"
tab to start.

[back to overview](#) > **Programming 1**[Course](#)[Content](#)[Lecturers](#)[Modes of delivery](#)[Programmes of study](#)[Examinations](#)**Examinations for this course**

the total percentage is not 100 %

And next on the "add"
button.

add

[back to overview](#) > [Programming 1](#) > **add/edit Examination**

Examination

Code	<input type="text" value="E166707032012153300"/>	If empty, a code
Description	<input type="text"/>	
Examination type	<input type="text" value="-choose-"/>	
Number of attempts	<input type="text" value="-choose-"/>	
Weight	<input type="text" value="0"/> %	
Active	<input type="text" value="yes"/>	
<input type="button" value="Submit"/>		

You now get a screen where you can define an examination event on the 2nd level (final exam or continuous assessment for instance).

Follow the example on the screen.

[back to overview](#) > [Programming 1](#) > add/edit Examination**Examination**

Code	<input type="text" value="E166707032012153300"/>	If empty, a code is generated (which can be overruled)
Description	<input type="text" value="Final Exam"/>	
Examination type	<input type="text" value="oral"/>	
Number of attempts	<input type="text" value="-choose-"/>	
Weight	<input type="text" value="0"/> %	
Active	<input type="text" value="yes"/>	
<input type="button" value="Submit"/>		

[back to overview](#) > [Programming 1](#) > **add/edit Examination**

Examination

Code	<input type="text" value="E166707032012153300"/>	If empty, a code is generated (which can be overruled)
Description	<input type="text" value="Final Exam"/>	
Examination type	<input type="text" value="oral"/>	
Number of attempts	<input type="text" value="2"/>	
Weight	<input type="text" value="50"/> %	
Active	<input type="text" value="yes"/>	
	<input type="button" value="Submit"/>	

The "Weight" is the relative importance of this examination event in the total or end result of the course.

[back to overview](#) > [Programming 1](#) > add/edit Examination

Examination

Code	<input type="text" value="E166707032012153300"/>	If empty, a code is generated (which can be overruled)
Description	<input type="text" value="Final Exam"/>	
Examination type	<input type="text" value="oral"/>	
Number of attempts	<input type="text" value="2"/>	
Weight	<input type="text" value="50"/> %	
Active	<input type="text" value="yes"/>	
<input type="button" value="Submit"/>		

Examination supervisors

There are no examination supervisors chosen. You must add at least one.

Tests

the total percentage is not 100 %

For every examination event there HAS TO BE a "supervisor" registered. Otherwise it will not be possible to enter marks for the examination event.

add

Click on the "add" button to add a supervisor. This may be the teacher of the course, but can also be another staff member.

[back to overview](#) > [Programming 1](#) > [Final Exam](#) > add/edit Examination supervisors**Examination supervisor****Second level unit**

Department of Computer Sciences (Parent ▼)

Lecturer

-choose-

Active

-choose-

Jones, Mary

Katucha, Allan

[back to overview](#) > [Programming 1](#) > add/edit Examination

Examination

Code	<input type="text" value="E166707032012153300"/>	If empty, a code is generated (which can be overruled)
Description	<input type="text" value="Final Exam"/>	
Examination type	<input type="text" value="oral"/>	
Number of attempts	<input type="text" value="2"/>	
Weight	<input type="text" value="50"/> %	
Active	<input type="text" value="yes"/>	
<input type="button" value="Submit"/>		

Examination supervisors

Description	Active
Allan Katucha	Y



Tests

the total percentage is not 100 %

[back to overview](#) > [Programming 1](#) > add/edit Examination

Examination

Code	<input type="text" value="E166707032012153300"/>	If empty, a code is generated (which can be overruled)
Description	<input type="text" value="Final Exam"/>	
Examination type	<input type="text" value="oral"/>	
Number of attempts	<input type="text" value="2"/>	
Weight	<input type="text" value="50"/> %	
Active	<input type="text" value="yes"/>	
<input type="button" value="Submit"/>		

Now we have added the first examination event, including the supervisor.

Examination supervisors

Description	Active
Allan Katucha	Y

Tests

the total percentage is not 100 %

Don't worry about this message here. It refers to the underlying (3rd) level. If you do not use this level, this message has no meaning. For this event, we do not define underlying tests or other events.

[back to overview](#) > [Programming 1](#) > add/edit Examination

Examination

Code If empty, a code is generated (which can be overruled)

Description

Examination

Number of a

Weight %

Active

Examination supervisors

Description	Active
Allan Katucha	Y



Tests

the total percentage is not 100 %

[back to overview](#) > **Programming 1**
[Course](#) | [Content](#) | [Lecturers](#) | [Modes of delivery](#) | [Programmes of study](#) | [Examinations](#)
Examinations for this course

the total percentage is not 100 %

Code	Description	Weight
E1667070320121	Final Exam	50

This message is relevant: we only have defined one examination event up to now (2nd level) which only counts for 50% of the course result.

So we now will add a second event, counting for the other 50%, the periodic testing or "Continuous assessment" part. (of course the percentages an event counts for is up to you to decide)

add



[back to overview](#) > [Programming 1](#) > **add/edit Examination****Examination**

Code	<input type="text" value="E166707032012153336"/>	If empty, a code is generated (which can be overruled)
Description	<input type="text"/>	
Examination type	<input type="text" value="-choose-"/>	
Number of attempts	<input type="text" value="-choose-"/>	
Weight	<input type="text" value="0"/> %	
Active	<input type="text" value="yes"/>	
<input type="button" value="Submit"/>		

[back to overview](#) > [Programming 1](#) > add/edit Examination**Examination**

Code If empty, a code is generated (which can be overruled)

Description

Examination type

Number of attempts

Weight %

Active

Examination supervisors

There are no examination supervisors chosen. You must add at least one.

Tests

the total percentage is not 100 %

[back to overview](#) > [Programming 1](#) > **add/edit Examination**

Examination

Code If empty, a code is generated (which can be overruled)

Description

Examination type

Number of attempts

Weight %

Active

Examination supervisors

There are no examination supervisors chosen. You must add at least one.

Now we register the supervisor.



Tests

the total percentage is not 100 %

[back to overview](#) > [Programming 1](#) > [Continuous assessment](#) > add/edit Examination supervisors[add](#)**Examination supervisor****Second level unit**

Department of Computer Sciences (Parent ▼)

Lecturer

Katucha, Allan ▼

Active

yes

[back to overview](#) > [Programming 1](#) > add/edit Examination

Examination

Code	<input type="text" value="E166707032012153336"/>	If empty, a code is generated (which can be overruled)
Description	<input type="text" value="Continuous assessment"/>	
Examination type	<input type="text" value="continuous assessment"/>	
Number of attempts	<input type="text" value="1"/>	
Weight	<input type="text" value="50"/> %	
Active	<input type="text" value="yes"/>	
<input type="button" value="Submit"/>		

Examination supervisors

Description	Active
Allan Katucha	Y



Tests

the total percentage is not 100 %

[back to overview](#) > [Programming 1](#) > add/edit Examination

Examination

Code	<input type="text" value="E166707032012153336"/>	If empty, a code is generated (which can be overruled)
Description	<input type="text" value="Continuous assessment"/>	
Examination type	<input type="text" value="continuous assessment"/>	
Number of attempts	<input type="text" value="1"/>	
Weight	<input type="text" value="50"/> %	
Active	<input type="text" value="yes"/>	
<input type="button" value="Submit"/>		

Examination supervisors

Description	Active
Allan Katucha	Y

Tests

the total percentage is not 100 %

Now, the "Continuous assessment" consists of a number of "underlying" tests (level 3), that we will define now. Click here to add the first test.

[add](#)
[add](#)

[back to overview](#) > [Programming 1](#) > [Continuous assessment](#) > **add/edit Test**

Test

Code	<input type="text" value="T21807032012153409"/>	If empty, a code is generated (which can be overruled)
Description	<input type="text" value="Test 1"/>	
Examination type	<input type="text" value="written"/>	
Number of attempts	<input type="text" value="1"/>	
Weight	<input type="text" value="50"/> %	
Active	<input type="text" value="yes"/>	
	<input type="button" value="Submit"/>	

This percentage now indicates how much this test (level 3) counts for in the total of the continuous assessment (level 2).

[back to overview](#) > [Programming 1](#) > [Continuous assessment](#) > **add/edit Test****Test**

Code	<input type="text" value="T21807032012153409"/>	If empty, a code is generated (which can be overruled)
Description	<input type="text" value="Test 1"/>	
Examination type	<input type="text" value="written"/>	
Number of attempts	<input type="text" value="1"/>	
Weight	<input type="text" value="50"/> %	
Active	<input type="text" value="yes"/>	

[back to overview](#) > [Programming 1](#) > [Continuous assessment](#) > **add/edit Test**

Test

Code If empty, a code is generated (which can be overruled)

Description

Examination type

Number of attempts

Weight %

Active

tests supervisors

Description

Active

Also tests (or other events on level 3) have to have a supervisor.



[back to overview](#) > [Programming 1](#) > [Continuous assessment](#) > **add/edit Test**

Test

Code If empty, a code is generated (which can be overruled)

Description

Examination type

Number of attempts

Weight %

Active

tests supervisors

Description

Active

Allan Katucha

Y



[back to overview](#) > [Programming 1](#) > **add/edit Examination**

Examination

Code	<input type="text" value="E166707032012153336"/>	If empty, a code is generated (which can be overruled)
Description	<input type="text" value="Continuous assessment"/>	
Examination type	<input type="text" value="continuous assessment"/>	
Number of attempts	<input type="text" value="1"/>	
Weight	<input type="text" value="50"/> %	
Active	<input type="text" value="yes"/>	
<input type="button" value="Submit"/>		

Examination supervisors

Description	Active	
Allan Katucha	Y	<input type="button" value="X"/>

Tests

the total percentage is not 100 %

Code	Description	Weight
T21807032012153409	Test 1	50



Since we only have 1 test which only counts for 50% of the Continuous assessment, we still have to add at least another one.

[back to overview](#) > [Programming 1](#) > [Continuous assessment](#) > add/edit Test**Test**

Code If empty, a code is generated (which can be overruled)

Description

Examination type

Number of attempts

Weight %

Active

tests supervisors**Description**

Active

Notice that we forget to
define a supervisor.....

Courses overview

Second level unit

Department of Computer Sciences (Parent ▼)

Study

-choose- ▼

Academic year

-choose- ▼

Code	Name	Primary study	Academic year
S007032012153048	Programming 1	Computer Science	2013

Total number of Courses: 1 page 1 of 1 1

We now have all the information for the course registered, so if we click on the edit button here, we can check the information we registered.

[back to overview](#) > Programming 1

Course

Content

Lecturers

Modes of delivery

Programmes of study

Examinations

Content

first press save-button in WYSIWYG widget, then press submit button

This module aims to make a start on developing skills to progress from an informal research question to a correct program. In addition to the syntax and semantics of several C constructions, special attention will be paid to problem-solving: how to convert a research question into an algorithm/program in C.

The Imperative Programming module forms, together with the Object-oriented Programming and Advanced Object-oriented Programming modules, a broad basic introduction to imperative and object-oriented programming. This first module will deal with common control structures in imperative languages (choice, repetition, recursion). Since this is the first module

Submit

[back to overview](#) > **Programming 1**[Course](#)[Content](#)[Lecturers](#)[Modes of delivery](#)[Programmes of study](#)[Examinations](#)**Lecturers for this course**[add](#)**Description****Active**

Allan Katucha

Y



[back to overview](#) > **Programming 1**[Course](#)[Content](#)[Lecturers](#)[Modes of delivery](#)[Programmes of study](#)[Examinations](#)**Modes of delivery for this course**[add](#)**Description****Active**

lab/practical

Y





lecture





Y



[back to overview](#) > **Programming 1**
[Course](#) | [Content](#) | [Lecturers](#) | [Modes of delivery](#) | [Programmes of study](#) | [Examinations](#)
programmes of study for this course
[add](#)

Description	Academic year	Cardinal time unit	Study time	Category of study	Rigidity	Major / Minor	Prerequisite courses	Active	
Computer Science - Bachelor of science	2013	1	Daytime	regular programme	compulsory	major		Y	 

[back to overview](#) > **Programming 1**
[Course](#) | [Content](#) | [Lecturers](#) | [Modes of delivery](#) | [Programmes of study](#) | [Examinations](#)
Examinations for this course
[add](#)

Code	Description	Weight	Active	
E166707032012153300	Final Exam	50	yes	 
E166707032012153336	Continuous assessment	50	yes	 

[back to overview](#) > **Programming 1**
[Course](#) | [Content](#) | [Lecturers](#) | [Modes of delivery](#) | [Programmes of study](#) | [Examinations](#)
Examinations for this course

Code	Description	Weight
E166707032012153300	Final Exam	50
E166707032012153336	Continuous assessment	50

add

But we forgot to enter a supervisor for the second test, so let's do that now anyway.



[back to overview](#) > [Programming 1](#)> [Continuous assessment](#) > [Test 2](#) > add/edit Examination supervisors

add

tests supervisor

Second level unit

Department of Computer Sciences ▼

Lecturer

Katucha, Allan ▼

Active

yes ▼

Submit

[back to overview](#) >

[Studies](#)
[Courses](#)
[Transfer curriculum](#)
[Academic years](#)
[Overview](#)
[Add](#)
[is assessment](#) > [add/edit Test](#)
Test
Code
Description
Examination type
Number of attempts
Weight
Active
tests supervisors
Description

Allan Katucha

Active

Y

If empty, a code is generated (which can be overruled)

Now, let's add a second course, but this time a course of the 2nd semester (the previous one was in the 1st semester).

[add](#)

[back to overview](#) > **new**

Course

Course
Second level unit

Department of Computer Sciences (Parent ▼

Primary study

Computer Science ▼

Code

S007032012155239

If empty, a code is generated (which can be overruled)

Name

Programming 2

Weight/Credit

2.0

The valid weight/credit format is 999.9

Study time

Daytime ▼

Max. no. of participants

0

Academic year

2013 ▼

Result type

Default ▼

Active

yes ▼

Submit

[back to overview](#) > **Programming 2**[Course](#) | [Content](#) | [Lecturers](#) | [Modes of delivery](#) | [Programmes of study](#) | [Examinations](#)**Course**

Second level unit

Department of Computer Sciences (Parent ▼)

Primary study

▼

Code

Name

Weight

Study time

Daytime ▼

Max. no. of participants

0

Academic year

2013 ▼

Result type

Default ▼

Active

yes ▼

If empty, a code is generated (which can be overruled)

The valid weight/credit format is 999.9

Defining the lecturer for the
course.

[back to overview](#) > [Programming 2](#) > add/edit Courses taught**Lecturer for this course****Second level unit**

Department of Computer Sciences (Parent ▼)

Lecturer

Jones, Mary ▼

Active

yes ▼

[back to overview](#) > **Programming 2**[Course](#)[Content](#)[Lecturers](#)[Modes of delivery](#)[Programmes of study](#)[Examinations](#)**Lecturers for this course**[add](#)**Description****Active**

Mary Jones

Y



[back to overview](#) > **Programming 2**
[Course](#) | [Content](#) | [Lecturers](#) | [Modes of delivery](#) | [Programmes of study](#) | [Examinations](#)
Lecturers for this course

Description

Mary Jones

Active

Y

add



Linking the course to our Programme
of Study (2nd semester)

(we do not show "Content" and "Modes
of delivery" registration for time's
sake).

[back to overview](#) > **Programming 2**[Course](#)[Content](#)[Lecturers](#)[Modes of delivery](#)[Programmes of study](#)[Examinations](#)**programmes of study for this course**

[back to overview](#) > [Programming 2](#) > add/edit Course / Programme of study

edit

After we submitted the information on the Programme of Study the course is part of, we see this line appearing. Here we can indicate whether there is a prerequisite course which the student should have taken before being able to take the course we are registering now.

Active

yes

Submit

Prerequisite courses

Code

Description

Academic year

Active

add

[back to overview](#) > [Programming 2](#) > **add/edit Course / Programme of study**

edit

Course / Programme of study

Second level unit Department of Computer Sciences (Parent ▼)

Course Programming 2

Academic year 2013

Study Computer Science ▼

Programme type Bachelor of science ▼

Category of study/Study time regular programme /Daytime ▼

Cardinal time unit number 2 ▼

Rigidity compulsory ▼

Major / Minor major ▼

Active yes ▼

Submit

Prerequisite courses

Code	Description	Academic year
------	-------------	---------------

In this case the 1st semester course "Programming 1" is a prerequisite to follow the course "Programming 2" we are actually registering. To indicate this we click on the "add" button here.

add

[back to overview](#) > [Programming 2](#) > [Computer Science](#) > add/edit Prerequisite course

edit

Prerequisite course

Second level unit Department of Computer Sciences (Parent ▼)
Study Computer Science ▼
Academic year 2013 ▼
Course Programming 1 (2013) ▼
Active yes ▼

Submit

Now we define that
"Programming 1" out of the
academic year 2013 as a
prerequisite course for
"Programming 2" in the
academic year 2013.

[back to overview](#) > [Programming 2](#) > add/edit Course / Programme of study

edit

Course / Programme of study

Second level unit

Department of Computer Sciences (Parent ▼)

Course

Programming 2

Academic year

2013

Study

Computer Science ▼

Programme type

Bachelor of science ▼

Category of study/Study time

regular programme /Daytime ▼

Cardinal time unit number

▼

Rigidity

▼

Major / Minor

▼

Active

▼

The course "Programming 1" is now in the list of prerequisite courses for "Programming 2".

Prerequisite courses

Code

Description

Academic year

Active

S007032012153048

Programming 1

2013

Y

add



[back to overview](#) > **Programming 2**[Course](#)[Content](#)[Lecturers](#)[Modes of delivery](#)[Programmes of study](#)[Examinations](#)**Examinations for this course**

the total percentage is not 100 %

[add](#)

Finally, and to finish this instruction, we define the examination structure for the course "Programming 2".

[back to overview](#) > [Programming 2](#) > **add/edit Examination****Examination**

Code	<input type="text" value="E166807032012155352"/>	If empty, a code is generated (which can be overruled)
Description	<input type="text" value="Final exam"/>	
Examination type	<input type="text" value="lab/practical"/>	
Number of attempts	<input type="text" value="2"/>	
Weight	<input type="text" value="100"/> %	
Active	<input type="text" value="yes"/>	
	<input type="button" value="Submit"/>	

This time we only define one examination event "Final exam" which counts for 100% of the course result.

[back to overview](#) > [Programming 2](#) > add/edit Examination**Examination**

Code	<input type="text" value="E166807032012155352"/>	If empty, a code is generated (which can be overruled)
Description	<input type="text" value="Final exam"/>	
Examination type	<input type="text" value="lab/practical"/>	
Number of attempts	<input type="text" value="2"/>	
Weight	<input type="text" value="100"/> %	
Active	<input type="text" value="yes"/>	
<input type="button" value="Submit"/>		

Examination supervisors

There are no examination supervisors chosen. You must add at least one.

Tests

the total percentage is not 100 %

[back to overview](#) > [Programming 2](#) > add/edit Examination

Examination

Code	<input type="text" value="E166807032012155352"/>	If empty, a code is generated (which can be overruled)
Description	<input type="text" value="Final exam"/>	
Examination type	<input type="text" value="lab/practical"/>	
Number of attempts	<input type="text" value="2"/>	
Weight	<input type="text" value="100"/> %	
Active	<input type="text" value="yes"/>	
<input type="button" value="Submit"/>		

Examination supervisors

There are no examination supervisors chosen. You must add at least one.



Tests

the total percentage is not 100 %



[back to overview](#) > [Programming 2](#) > [Final exam](#) > add/edit Examination supervisors[add](#)**Examination supervisor****Second level unit**

Department of Computer Sciences (Parent ▼)

Lecturer

Jones, Mary ▼

Active

yes ▼

[back to overview](#) > [Programming 2](#) > add/edit Examination

Examination

Code	<input type="text" value="E166807032012155352"/>	If empty, a code is generated (which can be overruled)
Description	<input type="text" value="Final exam"/>	
Examination type	<input type="text" value="lab/practical"/>	
Number of attempts	<input type="text" value="2"/>	
Weight	<input type="text" value="100"/> %	
Active	<input type="text" value="yes"/>	
<input type="button" value="Submit"/>		

Examination supervisors

Description	Active
Mary Jones	Y



Tests

the total percentage is not 100 %

[back to overview](#) > [Programming 2](#) > add/edit Examination

Click here to see an overview of the Courses just entered.

If empty, a code is generated (which can be overruled)

 %

Weight
Active
Examination supervisors

Description
Active

Mary Jones

Y


Tests

the total percentage is not 100 %

Courses overview

Second level unit

Department of Computer Sciences (Parent ▼)





Study

-choose- ▼

Academic year

-choose- ▼

Code	Name	Primary study	Academic year	Study time	Assigned	Active	
S007032012153048	Programming 1	Computer Science	2013	Daytime	no	Y	 
S007032012155239	Programming 2	Computer Science	2013	Daytime	no	Y	 

Total number of Courses: 2 page 1 of 1 1





Courses overview

Se
St
Ac

Department of Computer Sciences (Parent ▼
▼
▼

We now have two courses
defined, offered by our
Department and ...

Code	Name	Primary study	Academic year	Study time	Assigned	Active	
S007032012153048	Programming 1	Computer Science	2013	Daytime	no	Y	 
S007032012155239	Programming 2	Computer Science	2013	Daytime	no	Y	 

Total number of Courses: 2 page 1 of 1 1

Courses overview
Second level unit
Study
Academic year
Studies

Courses

Transfer
curriculum

Academic

year

Overview

Add

part-time

choose-

Computer Sciences (Parent ▼

... linked to our Programme of Study (BSc Computer Science), as is shown when we go to the "fact sheet" of that Programme of Study (to go there, follow the example on the screen).

Code

S007032012153048

S007032012155239

Total number of Courses: 2



year	Study time	Assigned	Active
	Daytime	no	Y
	Daytime	no	Y

<< >>



 
 

<< >>

Studies overview
Second level unit

Department of Computer Sciences (Parent ▼)

				<< >>	
Second level unit	Id	Name	Active		
Department of Computer Sciences	1210	Computer Science	yes		

Total number of Studies: 1 page 1 of 1 1

<< >>

[back to overview](#) > **Computer Science**

Study Address(es)

Study
Second level unit

Department of Computer Sciences (Parent ▼)

Name

Computer Science

Academic field

Computer Science ▼

Date of establishment

month year

The valid date format is MM-yyyy

09 2011

Start date

month year

The valid date format is MM-yyyy

01 2012

Active

yes ▼

Submit

Grade types for this study

add

**Programme of
study type**
Academic year
Study time
**Category of
study**
Study intensity
**Number of
cardinal time
units**
Contact
Active
[Computer
Science -
Bachelor of
science](#)

2013

Daytime

regular
programme

fulltime

6

Y



Discipline group	<div>-choose-</div>
Contact	<div>Jones, Mary</div>
Active	<div>yes</div>

Elective courses

Cardinal time unit	Elective courses
semester 1	<div>0</div>
semester 2	<div>0</div>
semester 3	<div>0</div>
semester 4	<div>0</div>
semester 5	<div>0</div>
semester 6	<div>0</div>

As you can see the courses are now in the list on the fact sheet of the Programme of Study.

Repeat the instruction in this video for every course to build the total list of courses for a programme.

Secondary school subjects (r)	<div>add</div>
Minimum number of subjects to grade	
3	Science, <div><div></div><div>×</div></div>
2	Computer science <div><div></div><div>×</div></div>

Course / Programme of study					<div>add</div>
Code	Course	semester	Rigidity	Major / Minor	
S007032012153048	Programming 1	1	compulsory	major	<div>×</div>
S007032012155239	Programming 2	2	compulsory	major	<div>×</div>

Prerequisite programmes of study		<div>add</div>
Study	Programme of study type	