

**ASSESSMENT REPORT
CENTRAL NEW MEXICO COMMUNITY COLLEGE**

The purpose of this form is to provide a written summary of your assessment results for the current assessment cycle.

Fall, 2011 – Spring 2012
 (Assessment Period Covered)

June 20, 2012
 (Date Report Submitted)

Choose ONE of the following 3 areas for this assessment report and insert the name of the general education area, certificate, degree or discipline on the appropriate line:

See definitions for each category in Assessment Process document

Gen Ed Area (see definitions)	or	Program	Computer Information Systems, Computer Programming Concentration
AA/AS <input type="checkbox"/> AAS <input type="checkbox"/>		Certificate AA/AS AAS	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>
Or Discipline Area (see definitions)			

Outcome(s) assessed:

1. Class construction: Write programs that contain a programmer-written class and demonstrate its use in the C++, Java and C# languages.
2. Class inheritance, and polymorphism: Write a program that contains a programmer-written class structure including a parent class and at least two children classes. The program must demonstrate polymorphism.
3. Graphical User Interface and Technical documentation: Write a program that contains a Graphical User Interface that includes event handling components. These components must include components such as menus, dialog boxes, sliders, buttons, and spinners. Tooltips must be on all components, where relevant. The program must contain a help section or additional documentation for the user.
4. Database manipulation and Web Application: Write a program that demonstrates the ability to connect to and manipulate a SQL database.
5. Web research: Use a search engine, such as "Google", to find information on classes or functions that are needed in a program. This web research includes finding the appropriate class/function, its documentation, and implementing the code in a program.
6. Debugging: Demonstrate the use of a debugging tool in at least two Integrated Development Environments, with at least two languages.
7. Linux: Students will demonstrate how to install, configure, create user accounts, issue correct commands and options, and perform standard network administration.

Classes/Cohort Assessed:

Outcomes 1-6 CIS Computer Programming students were assessed in their final semester via the CIS 2999 Capstone course.

Outcome 7: All CIS concentrations which require Linux in their program, will report Linux assessment results. This assessment information reflects all CIS students who take the Linux course.

Measurement tool(s):

Outcomes 1-5 were assessed via a portfolio of work created by the students. The students were given a detailed document describing the required computer programming topics. The document /portfolio topics were described and requested in order of the competencies. Students created a portfolio and each section addressed and demonstrated programming concepts.

Outcome 6 was assessed via a debugging exam given to students. The students were given a C++ and choice between Java and C# programs in which to debug.

Outcome 7 was assessed via the CIS 1680 Linux Essentials course. All students taking this course will be assigned a final project that encapsulates the exit competencies for this course.

Type of tool (for each tool listed above, indicate type of tool):

Outcomes 1-5: Direct (individual student performance) as demonstrated by program code and descriptions created by the students.

Outcome 6 (debugging): Direct (individual student performance) using an Internal tool (debugging exams created by two full-time computer programming faculty.)

Outcome 7: (Linux) Direct (individual student performance) using an Internal tool (debugging exams created by full-time network faculty.)

Achievement Target (if more than one measurement tool, list target for each tool separately):

Outcomes 1-6: The Computer Programming exit competencies are evaluated using a Rubrics with a scale of 4=excellent, 3=good, 2=fair and 1=poor. We believe a score of 3+ for 75% of our students represents success in accomplishing our goals.

Outcome 7: Several CIS concentrations incorporate the Linux course in its area of studies. Our achievement target for all Linux students (for all concentrations requiring this course) is 80%+ on the assessment skills exam for 75% of our students.

Assessment Results/Findings (if more than one measurement tool, list results for each tool separately):

Outcomes 1-6: A total of 20 Computer Programming students completed Computer Programming assessment activities in the Capstone course in Fall, 2011 and Spring 2012. Here are the results.

	COMP 1	COMP 2	COMP 3	COMP 4	COMP 5	COMP 6
SCORE	CLASSES	INHERITANCE	GUI	DB MANIP	RESEARCH	DEBUG
4	10	4	10	10	16	7
3.5	0	4	0	0	0	1
3	7	4	2	6	1	3
2.5	2	5	2	0	0	3
2.0	1	0	4	1	0	6
1.0	0	3	2	3	3	0

Using the Achievement Target of 3+ criteria for 75% of our students, the raw data is:

	COMP 1	COMP 2	COMP 3	COMP 4	COMP 5	COMP 6
SCORE	CLASSES	INHERITANCE	GUI	DB MANIP	RESEARCH	DEBUG
3+	17	12	12	16	17	11
<3	3	8	8	4	3	9
Meet Target?	Yes	No	No	Yes	Yes	No

Important Note: In Fall, 2011, as in previous semesters, we sat down with each student and had him/her talk to us, demonstrate the code and concepts and show us how to use a debugger in a development environment. In Spring, 2012, we required each student to produce a portfolio documenting the programming concepts through code he/she had written. The directions for this document were quite specific. Also, students were required to come into the lab and actually debug a program instead of just talking about using a debugger. (Very different results!)

Although students were being evaluated on the exact same concepts, the assessment approach was different. These two table illustrate our exit competency results in Fall versus Spring.

	FALL 2011					
	COMP 1	COMP 2	COMP 3	COMP 4	COMP 5	COMP 6
SCORE	CLASSES	INHERITANCE	GUI	DB MANIP	RESEARCH	DEBUG
3+	8	7	7	6	8	6
<3	0	1	1	2	0	2
Meet Target?	Yes	Yes	Yes	Yes	Yes	Yes

	SPRING 2012					
	COMP 1	COMP 2	COMP 3	COMP 4	COMP 5	COMP 6
SCORE	CLASSES	INHERITANCE	GUI	DB MANIP	RESEARCH	DEBUG
3+	9	5	5	9	9	5
<3	3	7	7	3	3	7
Meet Target?	Yes	No	No	Yes	Yes	No

We believe that the “true” results for our students are in the middle here. Assessing the students in the one-on-one conversation probably resulted in our being more lack in assessing how well the students really understood the material. (i.e., “OK, you’re on the right track”) Conversely, in the Spring, we found that some of the portfolios were lacking But, some students didn’t take the portfolio as seriously as necessary and some students who are very good programmers did not follow the direction on preparing the portfolio document.

(See Action Plan, in September Version for our corrections to this process.)

Outcome 7: Linux Results

A total of 109 students took the Linux skills exam. This encompasses 7 of the 8 sections (1 section the instructor did not give the exam).

SUCCESS SCORE	RAW TOTAL (OUT OF 109)	%
EXCELLENT 90-100 4	60	55.05
GOOD 80-89 3	18	16.51
FAIR 70-70 2	14	12.84
POOR 69 OR LESS	17	15.60
TOTAL SCORES 3 +	78/109	71.56

Meet target of 80%, score of 3 or 4 for 75% of our students? No.

Action Plan (close the loop):

**ASSESSMENT REPORT
CENTRAL NEW MEXICO COMMUNITY COLLEGE**

The purpose of this form is to provide a written summary of your assessment results for the current assessment cycle.

Fall, 2011 – Spring 2012
(Assessment Period Covered)

June 20, 2012
(Date Report Submitted)

Choose ONE of the following 3 areas for this assessment report and insert the name of the general education area, certificate, degree or discipline on the appropriate line:

See definitions for each category in Assessment Process document

Gen Ed Area (see definitions) _____ AA/AS <input type="checkbox"/> AAS <input type="checkbox"/>	or	Program _____ Certificate AA/AS AAS	Computer Information Systems, Computer Programming Concentration	_____ <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>
Or Discipline Area (see definitions) _____				

Outcome(s) assessed:

1. Class construction: Write programs that contain a programmer-written class and demonstrate its use in the C++, Java and C# languages.
2. Class inheritance, and polymorphism: Write a program that contains a programmer-written class structure including a parent class and at least two children classes. The program must demonstrate polymorphism.
3. Graphical User Interface and Technical documentation: Write a program that contains a Graphical User Interface that includes event handling components. These components must include components such as menus, dialog boxes, sliders, buttons, and spinners. Tooltips must be on all components, where relevant. The program must contain a help section or additional documentation for the user.
4. Database manipulation and Web Application: Write a program that demonstrates the ability to connect to and manipulate a SQL database.
5. Web research: Use a search engine, such as "Google", to find information on classes or functions that are needed in a program. This web research includes finding the appropriate class/function, its documentation, and implementing the code in a program.
6. Debugging: Demonstrate the use of a debugging tool in at least two Integrated Development Environments, with at least two languages.
7. Linux: Students will demonstrate how to install, configure, create user accounts, issue correct commands and options, and perform standard network administration.

Classes/Cohort Assessed:

Outcomes 1-6 CIS Computer Programming students were assessed in their final semester via the CIS 2999 Capstone course.

Outcome 7: All CIS concentrations which require Linux in their program, will report Linux assessment results. This assessment information reflects all CIS students who take the Linux course.

Measurement tool(s):

Outcomes 1-5 were assessed via a portfolio of work created by the students. The students were given a detailed document describing the required computer programming topics. The document /portfolio topics were described and requested in order of the competencies. Students created a portfolio and each section addressed and demonstrated programming concepts.

Outcome 6 was assessed via a debugging exam given to students. The students were given a C++ and choice between Java and C# programs in which to debug.

Outcome 7 was assessed via the CIS 1680 Linux Essentials course. All students taking this course will be assigned a final project that encapsulates the exit competencies for this course.

Type of tool (for each tool listed above, indicate type of tool):

Outcomes 1-5: Direct (individual student performance) as demonstrated by program code and descriptions created by the students.

Outcome 6 (debugging): Direct (individual student performance) using an Internal tool (debugging exams created by two full-time computer programming faculty.)

Outcome 7: (Linux) Direct (individual student performance) using an Internal tool (debugging exams created by full-time network faculty.)

Achievement Target (if more than one measurement tool, list target for each tool separately):

Outcomes 1-6: The Computer Programming exit competencies are evaluated using a Rubrics with a scale of 4=excellent, 3=good, 2=fair and 1=poor. We believe a score of 3+ for 75% of our students represents success in accomplishing our goals.

Outcome 7: Several CIS concentrations incorporate the Linux course in its area of studies. Our achievement target for all Linux students (for all concentrations requiring this course) is 80%+ on the assessment skills exam for 75% of our students.

Assessment Results/Findings (if more than one measurement tool, list results for each tool separately):

Outcomes 1-6: A total of 20 Computer Programming students completed Computer Programming assessment activities in the Capstone course in Fall, 2011 and Spring 2012. Here are the results.

	COMP 1	COMP 2	COMP 3	COMP 4	COMP 5	COMP 6
SCORE	CLASSES	INHERITANCE	GUI	DB MANIP	RESEARCH	DEBUG
4	10	4	10	10	16	7
3.5	0	4	0	0	0	1
3	7	4	2	6	1	3
2.5	2	5	2	0	0	3
2.0	1	0	4	1	0	6
1.0	0	3	2	3	3	0

Using the Achievement Target of 3+ criteria for 75% of our students, the raw data is:

	COMP 1	COMP 2	COMP 3	COMP 4	COMP 5	COMP 6
SCORE	CLASSES	INHERITANCE	GUI	DB MANIP	RESEARCH	DEBUG
3+	17	12	12	16	17	11
<3	3	8	8	4	3	9
Meet Target?	Yes	No	No	Yes	Yes	No

Important Note: In Fall, 2011, as in previous semesters, we sat down with each student and had him/her talk to us, demonstrate the code and concepts and show us how to use a debugger in a development environment. In Spring, 2012, we required each student to produce a portfolio documenting the programming concepts through code he/she had written. The directions for this document were quite specific. Also, students were required to come into the lab and actually debug a program instead of just talking about using a debugger. (Very different results!)

Although students were being evaluated on the exact same concepts, the assessment approach was different. These two table illustrate our exit competency results in Fall versus Spring.

	FALL 2011					
	COMP 1	COMP 2	COMP 3	COMP 4	COMP 5	COMP 6
SCORE	CLASSES	INHERITANCE	GUI	DB MANIP	RESEARCH	DEBUG
3+	8	7	7	6	8	6
<3	0	1	1	2	0	2
Meet Target?	Yes	Yes	Yes	Yes	Yes	Yes

	SPRING 2012					
	COMP 1	COMP 2	COMP 3	COMP 4	COMP 5	COMP 6
SCORE	CLASSES	INHERITANCE	GUI	DB MANIP	RESEARCH	DEBUG
3+	9	5	5	9	9	5
<3	3	7	7	3	3	7
Meet Target?	Yes	No	No	Yes	Yes	No

We believe that the “true” results for our students are in the middle here. Assessing the students in the one-on-one conversation probably resulted in our being more lack in assessing how well the students really understood the material. (i.e., “OK, you’re on the right track”) Conversely, in the Spring, we found that some of the portfolios were lacking But, some students didn’t take the portfolio as seriously as necessary and some students who are very good programmers did not follow the direction on preparing the portfolio document. (See Action Plan, in September Version for our corrections to this process.)

Outcome 7: Linux Results

A total of 109 students took the Linux skills exam. This encompasses 7 of the 8 sections (1 section the instructor did not give the exam).

SUCCESS SCORE	RAW TOTAL (OUT OF 109)	%
EXCELLENT 90-100 4	60	55.05
GOOD 80-89 3	18	16.51
FAIR 70-70 2	14	12.84
POOR 69 OR LESS	17	15.60
TOTAL SCORES 3 +	78/109	71.56

Meet target of 80%, score of 3 or 4 for 75% of our students? No.

Action Plan (close the loop):

Our capstone course has been reworked so that the students are required to meet 2-3 times with the mentor to be sure everyone understand what we are asking for their portfolio. The debugging test has proved a successful measure of debugging competency.

The Linux faculty are planning to evaluate the Linux skills exam to be sure it is an accurate measure of the student learning outcomes, and that the test is administered consistently across all sections. A new book has been selected for the course, which will aid in the skills preparation and exam procedure.