



# NORTHERN ARIZONA UNIVERSITY

## ECCC Proposal for New Course

1. Effective **BEGINNING** of what term and year?: Spring 2013  
*See effective dates calendar.*
2. College: Extended Campuses                      3. Academic Unit: Personalized Learning
4. Course subject and number: CIT 221                      . 5. Units: 3
6. Long course title: Programming Fundamentals  
*(max 100 characters including spaces)*
7. Short course title: Programming Fundamentals  
*(max. 30 characters including spaces)*
8. Catalog course description *(max. 60 words, excluding requisites):*

### **CIT 221 Programming Fundamentals**

An introduction to the fundamentals of programming, including scripting language, algorithmic design, object-oriented concepts, contemporary software development methodologies, testing and logical structures for problem solving using software engineering technology.

9. Grading option:      Letter grade                       Pass/Fail                       Both

10. Co-convened with: \_\_\_\_\_ 10a. UGC approval date\*: \_\_\_\_\_  
(For example: ESE 450 and ESE 550)  
\*Must be approved by UGC before UCC submission, and both course syllabi must be presented

11. Cross-listed with: \_\_\_\_\_  
(For example: ES 450 and DIS 450)  
Please submit a single cross-listed syllabus that will be used for all cross-listed courses.

12. May course be repeated for additional units? Yes  No   
12a. If yes, maximum units allowed? n/a  
12b. If yes, may course be repeated for additional units in the same term? Yes  No

13. Prerequisites: n/a

14. Co requisites: n/a

15. Is this course in any plan (major, minor or certificate) or sub plan (emphasis or concentration)? Yes  No   
If yes, describe the impact and attach written responses from the affected academic units prior to college curricular submission.

16. Is there a related plan or sub plan proposal being submitted? Yes  No   
If no, explain.

17. Does this course include combined lecture and lab components? Yes  No   
If yes, note the units specific to each component in the course description above.

18. Does this course duplicate content of existing courses? Yes  No   
If yes, list the courses with duplicate material. If the duplication is greater than 20%, explain why NAU should establish this course.

19. Names of the current faculty qualified to teach this course: Jeannie L. Copley

20. Justification for new course.

This course is needed to support the curriculum of the new Personalized Learning degree: Bachelor of Business Administration – Computer Information Technology.

**Answer 21-22 for UCC/YCC only:**

21. Is this course being proposed for Liberal Studies designation? Yes  No   
If yes, forward this form along with the appropriate supporting documentation to the

Liberal Studies Committee.

22. Is this course being proposed for Diversity designation? Yes  No   
If yes, forward this form along with the appropriate supporting documentation to the Diversity Committee

\_\_\_\_\_  
Reviewed by Curriculum Process Associate Date

Approvals:

\_\_\_\_\_  
Department Chair/Unit Head (if appropriate)

14 Sept. 2012  
Date

\_\_\_\_\_  
Chair of college curriculum committee

12 Sept. 12  
Date

\_\_\_\_\_  
Dean of college

11/13/12  
Date

For Committee use only:

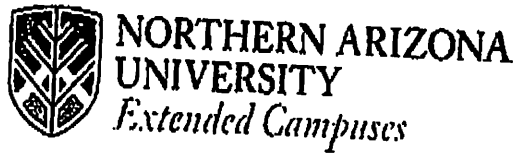
\_\_\_\_\_  
UCG/UGC/YCC Approval

11-7-12  
Date

Approved as submitted: Yes  No

Approved as modified: Yes  No

**Please attach proposed Syllabus in approved university format.**



## **PERSONALIZED LEARNING**

### **MASTER SYLLABUS - PROPOSED Programming Fundamentals (CIT 221)**

#### **I. Description:**

An introduction to the fundamentals of programming, including scripting language, algorithmic design, object-oriented concepts, contemporary software development methodologies, testing and logical structures for problems solving using software engineering technology.

#### **II. Student Learning Outcomes:**

- Understand and demonstrate proper graphical user interface design using: menus, controls, icons, dialogs, and color.
- Implement program solutions employing an appropriate development environment to build, debug, and test basic programs.
- Understand and explain the concept of object-oriented, event driven programming and how it differs from traditional programming.
- Demonstrate use of fundamental programming structures including sequence, selection, and repetition structures and the use of arrays.
- Implement program and database interaction (including a full update).
- Implement problem solutions using an appropriate and basic object-oriented programming language.
- Analyze business information system problems and develop appropriate Windows and Web-based solutions.
- Create appropriate documentation throughout the program development cycle.