Big Bend
COMMUNITY COLLEGE

MASTER COURSE OUTLINE

Prepared By: Arthur Wanner/Tom Willingham Date: April 2017

COURSE TITLE

Computer Programming Topics

GENERAL COURSE INFORMATION

Dept.: CS Course Num: 260 (Formerly:)
CIP Code: 11.0901 Intent Code: 21 Program Code: 527
Credits: 5
Total Contact Hrs Per Qtr.: 88
Lecture Hrs: 22 Lab Hrs: 66 Other Hrs:
Distribution Designation: General Elective (GE)

COURSE DESCRIPTION (as it will appear in the catalog)

This course highlights a new emerging software development, programming language, cloud computing, web application, or mobile application topic. In consultation with their Computer Science program advisor, students choose a specialized or in-depth programming related project and apply new and emerging computing and information technologies. Completed projects are presented and shared with fellow students.

PREREQUISITES

CS 111 or Instructor Permission

TEXTBOOK GUIDELINES

Textbook and materials to be determined by CS Faculty

COURSE LEARNING OUTCOMES

Upon successful completion of the course, students should be able to demonstrate the following knowledge or skills:

1. Identify new and emerging computing and information technology trends in programming languages
2. Expand fundamental programming skills
3. Apply skills learned in previous classes to create applications using current software languages

INSTITUTIONAL OUTCOMES

IO3 **Human Relations/Workplace Skills**: Students will be able to demonstrate teamwork, ethics, appropriate safety awareness and/or workplace specific skills

COURSE CONTENT OUTLINE

1. Determine project topic based on an industry trend in programming languages and student and advisor input. Suggested topics include, but are not limited to:
    - Mobile computing technologies
    - Agile software development
    - Programming and game development
    - Artificial intelligence for game developers

- Software testing and debugging
- Cloud computing and Software as a Service (SaaS)
- Independent study in a specific programming language such as Ruby, Ruby on Rails, PERL, Objective-C, etc.
- Enterprise software development
- Advanced programming topics related to a previously studied language, i.e.; C#, Java, C++, JavaScript, SQL, HTML, etc.

2. Develop project objectives and outcomes
3. Complete project
4. Presentations to fellow students

DEPARTMENTAL GUIDELINES *(optional)*
Student projects require pre-approval by the instructor.
Grading/Evaluation is based on
30% - Project quality
40% - Project breath, comprehensiveness, and completion
30% - Final presentations

____________________________________ ___________________

DIVISION CHAIR APPROVAL **DATE**