**Faculty Course Assessment Report**

**CS 3502, Operating Systems**

**Term and in-class/online designation/section**

*(Things in italics are instructions – not to be part of final FCAR)*

**Faculty Name: Kun Suo**

**Student Outcomes**

1. Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.
2. Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program’s discipline.

**Student Outcomes (program outcomes) for other programs** *(if your course is taken by more than your own majors, you need to list the student outcomes for each program that takes your course – so you could have Student Outcomes for CS, IT, SWE and CGDD for CSE 1321, etc. This will be prepopulated.)*

**SWE**

(m) An ability to appropriate apply discrete mathematics, probability and statistics, and relevant topics in computer science and support disciplines to complex software systems

**Course Outcomes**

At the end of the course students will be able to:

1. Demonstrate the understanding of the conceptual workings and design fundamentals of operating systems.
2. Demonstrate the understanding and be able to evaluate the features offered by various types of operating systems.
3. Demonstrate the understanding and be able to evaluate performance issues of computer systems.
4. Write short reports on the performance of various aspects of operating systems.

**Types of Assessments Used:** *(T for test, TQ for test questions, Q for quizzes, L/P for labs/projects, D for discussions, F for final, etc. – you will use this shorthand in your assessment results matrix)*

1. Two exams: T1, F

2. Two projects: P1, P2

3. Two homework: H1, H2

**Performance Criteria Level:** *Students will attain 75% for all assessments used to evaluate the attainment of the learning outcomes. (Either individual, or departmental performance criteria levels should be used here. It is better to have the department faculty choose the same performance criteria level for all courses – undergraduate at one level, and graduate at another level – such as 75% or undergraduate, and 83% for graduate, as an example.)*

**Assessment Data Matrix for Course Outcomes:** *(CO – course outcomes pre-populated above, T1, T2, L1/P1, Q3T1, etc. from assessment types used) – you will be putting down the averages of ALL students in the class (not individual students). You will be “mapping” your assessments used to support achievement of each course outcome. You have the option of using as your “n” students who completed the assessment only, or all students still in the class. If you use all students in the class, your average on achievement of the course outcome will be skewed.*

**Assessments to Course Outcomes Matrix: (Computer Science Majors)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| CO | H1Q14 (scheduling) | H2Q6  (page replacement) | P1  (system call) | P2 (pthread) | T1 | F |
| 1 | 80% |  | 83% |  | 75% |  |
| 2 |  | 91% |  |  |  | 85% |
| 3 |  |  |  |  |  | 85% |
| 4 |  |  |  | 80% |  |  |

**Analysis of Course Outcomes Assessment Results and Planned Improvements Based On Analysis:**

*Language C and how to use VM and terminal should be introduced ahead of this class. The first step to help student learn OS is step by step installing Linux VM.*

*The grader can provide more coding and debugging feedback to the students and open more hands-on labs for the projects. I will adopt Microsoft teams to interact with students in the future lectures.*

**Student Outcomes**

**Assessment Data Matrix for Student Outcomes:** *(these are the SOs listed at the top of the form – pre-populated; your percentages will be the same as those from your course outcomes matrix. – you will be putting down the averages of students in the class (not individual students). Again, I would suggest using students who completed the assessment chosen – not all students in the class. You don’t need to put down all assessments as you did for your course outcomes. Instead – you should have 1-2 assessments from your course (picked from the list of assessments used in the Course Outcomes) mapped to each Student Outcome. In fact, a good way to proceed would be to use particular questions from your final exam (if you have one), or questions from all exams as your assessments that meet your particular Student Outcomes associated with the course. If you have primarily projects (labs) in the course, you can, of course, use those as well. It’s your decision as to what particular assessment will meet the Student Outcomes.*

**Assessments to Student Outcomes Matrix:** *(these are the SOs listed at the top of the form – pre-populated; your percentages will be the same as those from your course outcomes matrix)*

|  |  |  |
| --- | --- | --- |
| **SO** | **FQ2** | **FQ3** |
| 1 | 83% |  |
| 2 |  | 85% |

**Assessments to Student Outcomes Matrix for other majors (CS, SWE, IT, CGDD):** *(these are the SOs listed at the top of the form – pre-populated for the different majors who take your course. DO NOT include if only your majors take the course.)*

**Improvements for course from Prior Term Brought Forward:**

*I will update project 1 to ask students installing Linux VM step by step. To make virtual environment stable, I will ask all students installing Linux on their on their local laptop. For the assignment, I will change the submission to report including design, code and screenshot.*

**Number of Students in the Course (after withdrawal date):** 25

**Course Grade Distribution – based on xx students in the course:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **A** | **B** | **C** | **D** | **F** | **W** | **I** |
| 52% | 24% | 20% | 0% | 4% | 0% | 0% |

**Other** *(anything else you would like to add about the course)*