College of San Mateo
Official Course Outline

1. **COURSE ID:** CIS 402 **TITLE:** Introduction to Artificial Intelligence
   Units: 4.0 units **Hours/Semester:** 48.0-54.0 Lecture hours; 48.0-54.0 Lab hours; 96.0-108.0 Homework hours; 192.0-216.0 Total Student Learning hours
   Method of Grading: Grade Option (Letter Grade or Pass/No Pass)
   Prerequisite: CIS 400

2. **COURSE DESIGNATION:**
   Degree Credit
   Transfer credit: CSU; UC
   AA/AS Degree Requirements:
   CSM - GENERAL EDUCATION REQUIREMENTS: E2b. Communication and Analytical Thinking

3. **COURSE DESCRIPTIONS:**
   Catalog Description:
   Ideas and techniques underlying the design of intelligent computer systems. Topics include search, game playing, knowledge representation, inference, planning, reasoning under uncertainty, machine learning, robotics, perception, and language understanding.

4. **STUDENT LEARNING OUTCOME(S) (SLO'S):**
   Upon successful completion of this course, a student will meet the following outcomes:
   1. Describe the fundamental concepts and techniques used in AI, including machine learning, neural networks, and natural language processing.
   2. Identify the key applications of AI in various industries and domains, including healthcare, finance, and autonomous vehicles.
   3. Analyze the strengths and limitations of different AI approaches and models, such as supervised and unsupervised learning.
   4. Develop basic AI algorithms and models using programming languages and tools commonly used in the field, such as Python and TensorFlow.
   5. Explain how AI can be used to solve real-world problems and contribute to innovation and creativity in various fields.
   6. Apply knowledge of AI to identify opportunities and challenges in different industries and propose solutions using AI techniques.

5. **SPECIFIC INSTRUCTIONAL OBJECTIVES:**
   Upon successful completion of this course, a student will be able to:
   1. Define key concepts and terminology of AI, including machine learning, neural networks, natural language processing, and expert systems.
   2. Describe the historical context and evolution of AI, including major breakthroughs, controversies, and ethical considerations.
   3. Explain the fundamental principles and techniques of AI, including search algorithms, logic, probabilistic reasoning, and decision trees.
   4. Apply AI techniques to solve real-world problems, such as image recognition, speech processing, and game playing.
   5. Evaluate the performance of AI algorithms and systems based on various metrics such as accuracy, efficiency, and scalability.
   6. Analyze the impact of AI on society and the workforce, and explore ethical and legal issues related to AI development and deployment.
   7. Collaborate effectively in teams to develop AI applications and systems.
   8. Communicate ideas and solutions clearly and effectively through written reports, oral presentations, and visual aids.

6. **COURSE CONTENT:**
   Lecture Content:
   - intelligent agents
   - uninformed search
   - informed search

- constraint satisfaction
- game-playing
- logical agents
- propositional logic
- first-order logic
- inference in first-order logic
- resolution, logic programming
- planning, plan execution
- uncertainty, probability theory, probabilistic inference
- Bayesian networks and associated inference algorithms
- optimal decisions under uncertainty
- optimal sequential decisions, Markov decision processes
- learning agents
- inductive learning, decision trees
- neural networks
- Bayesian learning
- natural language processing
- perception/vision
- robotics
- philosophical foundations

Lab Content:

1. Introduction to Python for AI: This lab could cover the basics of Python programming and how it is used in AI applications. Students could learn about data types, control structures, functions, and libraries commonly used in AI, such as numpy and pandas.
2. Search Algorithms: In this lab, students could implement various search algorithms, such as depth-first search, breadth-first search, and A* search, and compare their performance on different problems, such as navigating a maze or solving a puzzle.
3. Machine Learning Basics: This lab could introduce the basic concepts of machine learning, including supervised and unsupervised learning, training and testing data, and model evaluation. Students could use popular machine learning libraries, such as scikit-learn and TensorFlow, to implement simple models and classifiers.
4. Natural Language Processing: In this lab, students could learn how to use natural language processing techniques to analyze and manipulate text data. They could use libraries such as NLTK and spaCy to perform tasks such as tokenization, part-of-speech tagging, and sentiment analysis.
5. Reinforcement Learning: This lab could introduce the concept of reinforcement learning, where an agent learns to make decisions based on feedback from the environment. Students could implement simple reinforcement learning algorithms, such as Q-learning, and apply them to problems such as playing a simple game or controlling a simulated robot.
6. Project: In the final lab, students could work on a project of their choice, applying the concepts and techniques they have learned throughout the course. Projects could range from developing an AI-based game or chatbot to analyzing real-world data using machine learning techniques.

7. **REPRESENTATIVE METHODS OF INSTRUCTION:**

Typical methods of instruction may include:

A. Lecture
B. Lab
C. Activity
D. Directed Study
E. Discussion
F. Other (Specify): Lecture will be used to introduce new topics; Teacher will model problem-solving techniques. Class will solve a problem together, each person contributing a potential "step." Students will participate in short in-class projects (in teacher-organized small groups) to ensure that students experiment with the new topics in realistic problem settings; Teacher will invite questions AND ANSWERS from students, generating discussion about areas of misunderstanding; Teacher will create and manage an internet conference for discussion of course topics; and students will work in small groups on programming assignments.

8. **REPRESENTATIVE ASSIGNMENTS**

Representative assignments in this course may include, but are not limited to the following:

Writing Assignments:

In this course, students will mainly be tasked with creating documentation to support their lab and programming projects, encompassing both technical and end-user documentation. Technical documentation will cover the problem statement, project scope, solution overview, and any applicable limitations. On the other hand, end-user documentation will be provided to clients who use any reusable code.

Reading Assignments:

Students are expected to complete weekly textbook readings to enhance their understanding of the subject matter.

Other Outside Assignments:

The course assignments primarily consist of weekly exercises from the textbook and lab/programming assignments, which are designed to support the learning outcomes. Moreover, students will have to develop several significant programs that will entail writing 500-600 lines of code.

9. **REPRESENTATIVE METHODS OF EVALUATION**

Representative methods of evaluation may include:

A. Exams/Tests
B. Group Projects
C. Homework
D. Lab Activities
E. Projects
F. Quizzes
G. Written examination
H. Bi-weekly quizzes (short answer-from textbook material) to provide feedback to students and teacher; assessment of student contributions during class discussion and project time; individual programming assignments; Midterm and Final exams (short answer, general problem solving (similar to in-class work), short program segments (similar to programming assignments); Assessment of group participation on course projects, including peer assessment of participation and contribution to the group effort.

10. **REPRESENTATIVE TEXT(S):**

Possible textbooks include:

A. K. R. Chowdhary. *Fundamentals of Artificial Intelligence*, 1 ed. Springer Nature, 2020
B. Stuart Russell, Peter Norvig. *Artificial Intelligence: A Modern Approach*, 4 ed. Pearson Education, 2021

Origination Date: April 2023
Curriculum Committee Approval Date: September 2023
Effective Term: Fall 2024
Course Originator: Kamran Eftekhari