Artificial Intelligence

Certificate of Achievement

Proposed PSLOs

- Upon completion of the Certificate of Achievement in Artificial Intelligence, students will be able to analyze a given problem to determine which artificial intelligence algorithms are viable and apply their learned skills to develop an appropriate solution.
- Upon completion of the Certificate of Achievement in Artificial Intelligence, students will be able to use existing artificial intelligence and machine learning programming libraries on a given data set to create a valid model and justify their design decisions.

Artificial Intelligence - A CA - Certificate of Achievement (16 to fewer than 60 semester units)

Fall 2021

Course Sequence

Fall semester start only.

Required core.		**Units**
CS 1	Computing Fundamentals I	4
CS 7	Introduction to Computer Programming Concepts	3
MATH 40	Statistics and Probability	4
CS 2	Computing Fundamentals II	4
CS X3	Introduction to Artificial Intelligence	3
CS X4	Introduction to Machine Learning	3
Total Units		**21**

Program Mapping Template (Program: Artificial Intelligence)

□ AA □ AS □ AA-T □ AS-T ■ COA □ CC

This is an example course sequence. All plans can be modified to fit the needs of part-time students by adding more semesters. Students must meet with a counselor to complete an individualized educational plan.

SEMESTER 1 - FALL

Courses	Units	Semesters Offered	Prereq?	Advisory Notes
CS 1	4	Fall/Spring		CS 7 is strongly recommended for CS 1 and can be taken in the summer semester prior for better preparation of CS 1. MATH 40 has many possible ways to meet its prerequisite, including: MATH 50 or MATH 55 or MATH 55B or NMAT 250 or NMAT 255
CS 7	3	Summer/Fall/Spring		
MATH 40	4	Fall/Spring	Yes	
SEMESTER TOTAL	11			

SEMESTER 2 - SPRING

Courses	Units	Semesters Offered	Prereq?	Advisory Notes
CS 2	4	Fall/Spring	CS 1	
CS X3	3	Spring	CS 1,CS 7,MATH 40	
CS X4	3	Spring	CS 1,CS 7,MATH 40	
SEMESTER TOTAL	10			

SEMESTER 3 - SUMMER

Courses	Units	Semesters Offered	Prereq?	Advisory Notes
SEMESTER TOTAL				

SEMESTER 4 - FALL

Courses	Units	Semesters Offered	Prereq?	Advisory Notes
SEMESTER TOTAL				

SEMESTER 5 - SPRING

Courses	Units	Semesters Offered	Prereq?	Advisory Notes
SEMESTER TOTAL				
Total Major Coursework	17			
Total Units Required	21			

Las Positas CurricUNET

Las Positas College

Las Positas College
3000 Campus Hill Drive
Livermore, CA 94551-7650
(925) 424-1000
(925) 443-0742 (Fax)

Course Outline for CS X3

INTRODUCTION TO ARTIFICIAL INTELLIGENCE

Effective: Spring 2022

I. CATALOG DESCRIPTION:
CS X3 — INTRODUCTION TO ARTIFICIAL INTELLIGENCE — 3.00 units

An introduction to artificial intelligence (AI) and modern AI programming libraries. Basic discrete mathematics. Problem solving using uninformed, informed, local, and adversarial search algorithms. Knowledge representation, inference, and reasoning using propositional and first-order logic. Quantifying and reasoning about uncertainty with Bayesian networks and Markov decision processes. Ethical considerations of AI.

2.50 Units Lecture 0.50 Units Lab

Prerequisite
CS 7 - Introduction to Computer Programming Concepts
with a minimum grade of C

MATH 40 - Statistics and Probability
with a minimum grade of C

CS 1 - Computing Fundamentals I
with a minimum grade of C

Grading Methods:
Letter or P/NP

Discipline:
- Computer Science

	MIN
Lecture Hours:	45.00
Expected Outside of Class Hours:	90.00
Lab Hours:	27.00
Total Hours:	162.00

II. NUMBER OF TIMES COURSE MAY BE TAKEN FOR CREDIT: 1

III. PREREQUISITE AND/OR ADVISORY SKILLS:

Before entering the course a student should be able to:

A. CS7
1. Design simple algorithms to solve a variety programming problems.
2. Design and implement programs of short to medium length, using standard elements of programming languages such as variables, input/output, control structures, functions/methods and arrays.
3. Explain what an algorithm is and its importance in computer programming.
4. Analyze and investigate program behavior to effectively alter or debug existing code.
5. Design and implement specific program steps and components to achieve desired program behavior.

B. MATH40
1. Define different types of statistics, how they are used and misused;
2. Take real world raw data and organize it into tables, charts, and/or graphs both with and without the use of technology;
3. Interpret data displayed in tables and graphically;
4. Calculate and understand the meaning of the measures of central tendency: mean, median, mode, and the measures of variation and position: range, variance, and standard deviation as they relate to a discrete and continuous population, sample, or distribution;
5. Construct and interpret confidence intervals for single populations and two-populations comparisons;
6. Apply concepts of sample space and probability;
7. Determine the fundamentals concepts of probability and be able to calculate probabilities using some basic rules;
8. Perform descriptive and inferential statistics, using a software package (technology).

C. CS1
1. Create and interpret expressions involving arithmetic and logical operators;
2. Modify and expand short programs that use standard conditional and iterative control structures and functions.

3. Choose appropriate conditional and iteration constructs for a given programming task.
4. Apply the techniques of structured (functional) decomposition to break a program into smaller pieces.
5. Analyze and explain the behavior of simple programs.
6. Discuss and apply the concept of algorithms in problem-solving processes.
7. Judge the correctness and quality of algorithms, identifying necessary properties of good algorithms.
8. Design, implement, test, and debug programs using basic computation, simple I/O, standard conditional and iterative structures, and the definition of functions.

IV. MEASURABLE OBJECTIVES:

Upon completion of this course, the student should be able to:

A. Formulate an appropriate model for a well-defined problem by defining states, actions, a transition model, and goal testing.
B. Understand and explain what is artificial intelligence and how an agent can be considered intelligent and rational.
C. Design and implement problem-solving agents to solve search problems using appropriate search algorithms.
D. Develop and implement admissible and consistent heuristic functions for a search problem.
E. Design and implement knowledge-based agents that utilize propositional/first-order logic to infer and prove facts about the environment of the agent.
F. Design and implement decision-theoretic agents that select rational actions for a problem containing uncertainty.
G. Understand and reason about the underlying ethical issues in developing artificial intelligence.

V. CONTENT:

A. Intelligent agents
   1. Rationality
   2. Task environments
   3. Agent structure
B. Mathematical foundations
   1. Sets
   2. Functions
   3. Recursion
   4. Graphs
   5. Trees
C. Search in simple environments
   1. Formulating a well-defined problem
      a. States
      b. Actions
      c. Transition model
      d. Goal testing
   2. Uninformed graph algorithms
      a. Best-first search
      b. Breadth-first search
      c. Uniform-cost search
      d. Depth-first search
      e. Iterative deepening search
   3. Informed graph algorithms
      a. Heuristic functions
      b. Greedy best-first search
      c. A* search
D. Search in complex environments
   1. Hill-climbing search
   2. Simulated annealing
   3. Local beam search
   4. Genetic algorithms
E. Adversarial search in games
   1. Game theory
   2. Minimax algorithm
   3. Alpha-beta pruning
   4. Monte Carlo tree search
   5. Stochastic games
F. Logical agents
   1. Propositional logic
      a. Propositional theorem proving
      b. WalkSAT
   2. First-order logic
      a. Forward chaining
      b. Resolution theorem proving
   3. Knowledge engineering
G. Probabilistic reasoning
   1. Probabilistic inference
   2. Naïve Bayes models
   3. Bayesian networks
   4. Markov decision process
      a. Value iteration
      b. Policy iteration
H. Ethics of artificial intelligence

VI. METHODS OF INSTRUCTION:

A. **Audio-visual Activity** -
B. **Classroom Activity** -
C. **Demonstration** -
D. **Discussion** -
E. **Lecture** -
F. **Projects** -
G. **Written Exercises** -

VII. TYPICAL ASSIGNMENTS:

A. Give a PEAS description for different task environments, such as playing soccer or shopping.
B. Assume you are navigating a robot in a maze. Formulate the problem with a PEAS description, determine the state space, and perform both breadth-first search and depth-first search to find a path out of the maze. Implement this solution using Python.
C. Implement and test hill-climbing search in Python to solve the traveling salesperson problem.

D. For a game tree that is two moves deep, perform alpha-beta pruning and determine the minimax value of the root max node. How many nodes were pruned compared to the minimax algorithm?
E. For a 2-CNF propositional expression, prove using resolution that it entails a given knowledge base.
F. Model a simple, probabilistic grid environment in Python. Create an agent that uses policy iteration to find an optimal policy for a given start state.

VIII. EVALUATION:

Methods/Frequency

A. Exams/Tests

There should be at least two exams that each cover one half of the course content.

B. Projects

A semester project should be incorporated that allows students to apply the majority of the course content.

C. Class Participation

Students should be expected to participate in their learning environment, both during and outside of class hours. Participation can be graded weekly or bi-weekly.

D. Home Work

Homework assignments should be given on a weekly or bi-weekly basis. This should include written work and programming assignments.

IX. TYPICAL TEXTS:

1. Russell, Stuart, and Peter Norvig. *Artificial Intelligence: A Modern Approach.* 4 ed., Pearson, 2020.
2. Artasanchez, Alberto, and Prateek Joshi. *Artificial Intelligence with Python.* 2 ed., Packt, 2020.

X. OTHER MATERIALS REQUIRED OF STUDENTS:

A. Access to a computer with an active Internet connection

Las Positas CurricUNET

LAS POSITAS COLLEGE

Las Positas College
3000 Campus Hill Drive
Livermore, CA 94551-7650
(925) 424-1000
(925) 443-0742 (Fax)

Course Outline for CS X4

INTRODUCTION TO MACHINE LEARNING

Effective: Spring 2022

I. CATALOG DESCRIPTION:
CS X4 — INTRODUCTION TO MACHINE LEARNING — 3.00 units

An introduction to machine learning (ML), with an emphasis on programming ML applications and using modern ML libraries. Basic discrete mathematics and linear algebra. An overview of various supervised learning classifiers. Unsupervised learning via clustering. Reinforcement learning with model-based and model-free approaches. Safety and ethical concerns of ML.

2.50 Units Lecture 0.50 Units Lab

Prerequisite
CS 7 - Introduction to Computer Programming Concepts
with a minimum grade of C

MATH 40 - Statistics and Probability
with a minimum grade of C

CS 1 - Computing Fundamentals I
with a minimum grade of C

Grading Methods:
Letter or P/NP

Discipline:
- Computer Science

	MIN
Lecture Hours:	45.00
Expected Outside of Class Hours:	90.00
Lab Hours:	27.00
Total Hours:	162.00

II. NUMBER OF TIMES COURSE MAY BE TAKEN FOR CREDIT: 1

III. PREREQUISITE AND/OR ADVISORY SKILLS:

Before entering the course a student should be able to:

A. CS7
1. Design simple algorithms to solve a variety programming problems.
2. Design and implement programs of short to medium length, using standard elements of programming languages such as variables, input/output, control structures, functions/methods and arrays.
3. Explain what an algorithm is and its importance in computer programming.
4. Analyze and investigate program behavior to effectively alter or debug existing code.
5. Design and implement specific program steps and components to achieve desired program behavior.

B. MATH40
1. Define different types of statistics, how they are used and misused;
2. Identify the standard methods of obtaining data and identify the advantages and disadvantages of each;
3. Take real world raw data and organize it into tables, charts, and/or graphs both with and without the use of technology;
4. Interpret data displayed in tables and graphically;
5. Calculate and understand the meaning of the measures of central tendency: mean, median, mode, and the measures of variation and position: range, variance, and standard deviation as they relate to a discrete and continuous population, sample, or distribution;
6. Construct and interpret confidence intervals for single populations and two-populations comparisons;
7. Apply concepts of sample space and probability;
8. Determine the fundamentals concepts of probability and be able to calculate probabilities using some basic rules;
9. Perform descriptive and inferential statistics, using a software package (technology).
10. Formulating a hypothesis test by selecting the appropriate technique for testing the hypothesis and interpreting the result for one and two-populations comparisons;

11. Determine and interpret levels of statistical significance including p-values;

C. CS1
   1. Create and interpret expressions involving arithmetic and logical operators;
   2. Modify and expand short programs that use standard conditional and iterative control structures and functions.
   3. Choose appropriate conditional and iteration constructs for a given programming task.
   4. Apply the techniques of structured (functional) decomposition to break a program into smaller pieces.
   5. Analyze and explain the behavior of simple programs.
   6. Discuss and apply the concept of algorithms in problem-solving processes.
   7. Judge the correctness and quality of algorithms, identifying necessary properties of good algorithms.
   8. Design, implement, test, and debug programs using basic computation, simple I/O, standard conditional and iterative structures, and the definition of functions.

IV. MEASURABLE OBJECTIVES:

Upon completion of this course, the student should be able to:

A. Understand the reason for splitting data sets and be able to perform a cross-validation.
B. Understand, explain, and be able to identify the bias-variance tradeoff.
C. Select, implement, and use an appropriate classifier to conduct supervised learning with a given data set.
D. Implement a clustering algorithm to perform unsupervised learning on a data set.
E. Understand the difference between model-based and model-free reinforcement learning.
F. Design and implement a Q-learning agent for a reinforcement learning problem.
G. Understand and critically analyze the safety and ethical concerns of designing machine learning applications.

V. CONTENT:

A. Mathematical foundations
   1. Sets
   2. Functions
   3. Vectors
   4. Matrices

B. Supervised learning
   1. Model selection
      a. Training, validation, and test sets
      b. Cross-validation
      c. Hyperparameters
      d. Loss functions
   2. Bias-variance tradeoff
      a. Underfitting
      b. Overfitting
      c. Ockham's razor
      d. Noise
   3. k-nearest neighbor
      a. Curse of dimensionality
   4. Perceptron
   5. Naïve Bayes
      a. Maximum likelihood estimation
      b. Maximum a posteriori
   6. Logistic regression
   7. Linear regression
      a. Gradient descent
   8. Support vector machine
      a. Kernel functions
   9. Decision tree
      a. Entropy
      b. Information gain
   10. Ensemble learning
      a. Bagging
      b. Boosting
   11. Neural network
      a. Activation functions
      b. Back-propagation

C. Unsupervised learning
   1. Clustering

D. Reinforcement learning
   1. Markov decision process
      a. Value iteration
      b. Policy iteration
   2. Q-learning

E. Safety and ethics of machine learning
   1. Accountability
   2. Interpretability
   3. Explainability
   4. Sustainability

VI. METHODS OF INSTRUCTION:

A. **Audio-visual Activity** -
B. **Classroom Activity** -
C. **Demonstration** -
D. **Discussion** -
E. **Lecture** -
F. **Projects** -
G. **Written Exercises** -

VII. TYPICAL ASSIGNMENTS:

A. Given a data set, use Python to split the data set into training, validation, and test sets. Use 5-fold cross-validation.
B. Implement the perceptron classifier in Python. Test your implementation on a data set.
C. For a given corpus, use Python to generate two naïve Bayes models, one using MLE and the other with MAP. Visualize the error rate for both models against the test set. Why do you think one model works better than the other?
D. Design by hand a decision tree for a given small data set.
E. Design and implement a Q-learning agent in Python for the given Markov decision process.

VIII. EVALUATION:

Methods/Frequency

A. Exams/Tests

There should be at least two exams that each cover one half of the course content.

B. Projects

A semester project should be incorporated that allows students to apply the majority of the course content.

C. Class Participation

Students should be expected to participate in their learning environment, both during and outside of class hours. Participation can be graded weekly or bi-weekly.

D. Home Work

Homework assignments should be given on a weekly or bi-weekly basis. This should include written work and programming assignments.

IX. TYPICAL TEXTS:

1. Russell, Stuart, and Peter Norvig. *Artificial Intelligence: A Modern Approach.* 4 ed., Pearson, 2020.
2. Artasanchez, Alberto, and Prateek Joshi. *Artificial Intelligence with Python.* 2 ed., Packt, 2020.
3. Géron, Aurélien. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow.* 2 ed., O'Reilly, 2019.

X. OTHER MATERIALS REQUIRED OF STUDENTS:

A. Access to a computer with an active Internet connection