H9PAI: Programming for Artificial Intelligence

Module Code:	H9PAI
Long Title	Programming for Artificial Intelligence APPROVED
Title	Programming for AI
Module Level:	LEVEL 9
EQF Level:	7
EHEA Level:	Second Cycle
Credits:	10
Module Coordinator:	Arghir Moldovan
Module Author:	Arghir Moldovan
Departments:	School of Computing
Specifications of the qualifications and experience required of staff	MSc and/or PhD degree in computer science or cognate discipline. Experience lecturing in the field. May have industry experience also.

Learning Outcomes

On successful completion of this module the learner will be able to:

#	Learning Outcome Description
LO1	Analyse, compare, contrast and critically evaluate the characteristics of programming languages and environments commonly utilised for AI solutions implementation.
LO2	Critically assess the challenges associated with implementing AI solutions for various problems.
LO3	Critically assess methods and practices for software development to design and implement AI solutions requirements.
LO4	Evaluate, design and implement AI solutions by using key algorithms, data structures, and relevant programming languages.

Dependencies

Module Recommendations

No recommendations listed

Co-requisite Modules

No Co-requisite modules listed

Entry requirements	Internal to the programme

H9PAI: Programming for Artificial Intelligence

Module Content & Assessment

Indicative Content
Introduction to Programming for AI Module Introduction; History and evolution of programming languages used for AI; Programming types and paradigms (imperative, declarative, functional, logic, agent oriented programming, probabilistic programming, etc.); Overview of programming languages used for implementing AI solutions: general-purpose languages (e.g., Python), classical AI languages (e.g., Lisp), new generic AI programming languages (e.g., MIT Gen), deep probabilistic programming languages (e.g., Edward, Pyro).
AI Computation Challenges Challenges associated with big data requirements of statistical AI (e.g., deep learning); Computation challenges (e.g., search space, time and space complexity); Parallelism for computational processes; Use of specialised/dedicated hardware to speed up computations (e.g., GPUs, Google TPUs, wafer-scale AI chips such as Cerebras CS-1, etc.); Distributed computing platforms; Brief overview AI services and APIs offered by public cloud providers (e.g., Amazon AWS, Microsoft Azure, Google Cloud Platform).
Overview of the programming language Syntax and semantics, expressions and statements; Basic data types, conversion and coercion; Built in data structures (arrays, matrices, lists, etc.), indexing data structures; Program flow control and iteration.
Input/Output and Functions Input/output data from structured/semi-structured file formats (csv, xml, json); Input data from the Internet (e.g., web scraping, web APIs); Defining functions; Lambdas for functional programming; Algorithm design.
Data Operations and Data Streaming Dealing with missing values; Catching exceptions; Use of support libraries (e.g., Pandas, Numpy, dfply); Stream input sources, live data stream, window-based transformations, combination of batch and stream computations.
Database Programming – Relational Databases Database system technologies; Programmatically connecting to databases; Create/Read/Update/Delete (CRUD) Operations; SQL Optimization, Indexing and Normalization.
Database Programming – NoSQL Databases, Data Lakes NoSQL Overview and Data Models: Document Model, Key-Value Model, Column Family, Aggregates, Graph Model, Triple Stores; NoSQL Data Modelling Concepts; Query Languages for Data in NoSQL; NoSQL systems.
ETL, Data Pipelines and Data Wrangling Data wrangling techniques; Developing programs for data processing activities (e.g., data extraction, cleaning, merging, aggregation, validation, analysis, reporting).
Ontology Engineering Ontology definitions: domain ontology, concepts, instances and relationships. Overview of technologies for ontology engineering: Web Languages (e.g., HTML, XML and RDF), Metadata standards (e.g., Dublin Core), Ontology Language (e.g., OWL), Ontology Editor (e.g., Protégé), Reasoning language (e.g., SPARQL), reasoners (e.g., HermiT); Overview of Python packages for ontology-oriented programming (e.g., Owlready2, RDFlib, pyspaql, pronto, AllegroGraph).
Deep Learning A brief introduction to deep learning concepts; Overview of deep learning frameworks (e.g., PyTorch, TensorFlow, Apache MXNet, Keras); Overview of public cloud AI services for deep learning (e.g., AWS Deep Learning AMIs, Google Cloud TPUs); Use of pre-trained models and cloud services for various example applications (e.g., regression, classification).
Natural Language Processing Overview of NLP libraries and frameworks (e.g., NLTK); Overview of public cloud AI services for NLP, translation (e.g., Amazon Lex, Polly, etc.); Use of pre-trained Generalized Language Models for NLP applications (e.g., Google BERT, OpenAI GPT-2, etc.).
Image Processing Overview of image processing libraries and frameworks (e.g., OpenCV); Overview of public cloud AI services for image and video recognition (e.g., Azure Face, AWS Rekognition, etc.); Use of pre-trained models for example applications (e.g., RetinaNET object detection).

Assessment Breakdown	%
Coursework	100.00%

Assessments

Full Time

Coursework

Assessment Type:	Formative Assessment	**% of total:**	Non-Marked
Assessment Date:	n/a	**Outcome addressed:**	1,2,3,4
Non-Marked:	Yes		

Assessment Description:
Formative assessment will be provided on the in-class individual or group activities. Feedback will be provided in written or oral format, or on-line through Moodle. In addition, in class discussions will be undertaken as part of the practical approach to learning.

Assessment Type:	Continuous Assessment	**% of total:**	30
Assessment Date:	n/a	**Outcome addressed:**	3,4
Non-Marked:	No		

Assessment Description:
This assessment will consist of practical tasks in the form of an in-class test or homework. This will assess learners' knowledge and competences on core programming language concepts and operations covered so far.

Assessment Type:	Project	**% of total:**	70
Assessment Date:	n/a	**Outcome addressed:**	1,2,3,4
Non-Marked:	No		

Assessment Description:
The terminal assessment will consist of a project that will evaluate all learning outcomes. Learners will have to develop a software application of their own choice utilising appropriate AI programming languages, algorithms, techniques, tools / frameworks / services. The final submission will consist of a written report and the implemented AI solution artefact.

No End of Module Assessment

No Workplace Assessment

Reassessment Requirement

Coursework Only
This module is reassessed solely on the basis of re-submitted coursework. There is no repeat written examination.

Reassessment Description
The reassessment strategy for this module will consist of a project that will assess all learning outcomes. Students who fail the module will be afforded an opportunity to do the repeat project over the Summer months.

H9PAI: Programming for Artificial Intelligence

Module Workload

Module Target Workload Hours 0 Hours

Workload: Full Time

Workload Type	*Workload Description*	*Hours*	*Frequency*	*Average Weekly Learner Workload*
Lecture	Classroom & Demonstrations	24	Per Semester	2.00
Tutorial	Practical/Tutorial	24	Per Semester	2.00
Independent Learning	Independent learning	202	Per Semester	16.83
			Total Weekly Contact Hours	4.00

Workload: Blended

Workload Type	*Workload Description*	*Hours*	*Frequency*	*Average Weekly Learner Workload*
Lecture	Classroom & Demonstrations	12	Per Semester	1.00
Tutorial	Practical/Tutorial	12	Per Semester	1.00
Directed Learning	Directed Learning	24	Per Semester	2.00
Independent Learning	No Description	202	Per Semester	16.83
			Total Weekly Contact Hours	4.00

Module Resources

Recommended Book Resources

Artasanchez, A. & Joshi, P. (2020). Artificial Intelligence with Python(2nd ed.). Packt Publishing. [ISBN: 978-1839219535]..

Rothman, D., Lamons, M., Kumar, R., Nagaraja, A., Amir Ziai, & Dixit, A. (2018). Python: Beginner's Guide to Artificial Intelligence. Packt Publishing. [ISBN: 978-1789957327]..

Supplementary Book Resources

McKinney, W. (2017). Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython(2nd ed.). O'Reilly Media. [ISBN: 978-1491957660]..

Jean-Baptiste, L. (2020) Ontologies with Python: Programming OWL 2.0 Ontologies with Python and Owlready2. Apress. [ISBN: 978-1484265529]..

Bonaccorso, G., Fandango, A., & Shanmugamani, R. (2018). Python: Advanced Guide to Artificial Intelligence. Packt Publishing. [ISBN: 978-1789957211]..

This module does not have any article/paper resources

Other Resources

[Website], DataCamp, Learn R, Python & Data Science Online,,
https://www.datacamp.com/

Discussion Note:	