International Collegiate Artificial Intelligence Contest (ICAIC) Syllabus

[As at: 2026/07/20]

1. Topic Classifications and Expected Depth

ICAIC adopts the IOAI distinction between theoretical and practical knowledge and adds a depth level to support fair task setting at collegiate level.

Knowledge category

Category	Meaning
Theory (T)	Understand the main concepts, assumptions, mechanisms, and standard equations. Proof-heavy derivations are not expected unless explicitly stated.
Practice (P)	Use appropriate libraries and tools, implement or adapt methods, train and debug models, and interpret outputs.
Both (B)	Demonstrate both conceptual understanding and practical implementation ability.

Depth level

Level	Meaning
Core (C)	May be assessed directly without a task-specific tutorial.
Extended (E)	May be assessed when supported by an at-home task, starter material, provided implementation, or sufficient explanation in the task statement.

Reading the tables: an entry such as B/C means that both theory and practice are expected at Core level.

2. General Contestant Capabilities

Across all domains, contestants should be able to move from an unfamiliar dataset and task statement to a valid, reproducible, and progressively improved solution.

- Understand the task, data schema, target, constraints, evaluation metric, and submission format.
- Inspect data, identify quality issues, establish a simple baseline, and design a suitable validation strategy.
- Select methods that match the data modality, dataset size, compute budget, and evaluation objective.
- Implement, train, debug, profile, and compare models using the provided Python environment.
- Recognize leakage, overfitting, distribution shift, class imbalance, unstable training, and metric misuse.
- Interpret experiments and make evidence-based improvements under time and compute constraints.
- Produce deterministic outputs and correctly formatted submissions.

3. Mathematical, Statistical, and Optimization Foundations

Topic	Expected competency	Cat.	Lvl.
Linear algebra	Vectors, matrices, tensors, dot products, matrix multiplication, norms, rank, projections, eigenvalues/eigenvectors, and singular value decomposition.	B	C
Calculus and automatic differentiation	Derivatives, partial derivatives, gradients, chain rule, Jacobian/Hessian intuition, computational graphs, and backpropagation.	B	C

Probability	Random variables, common distributions, expectation, variance, covariance, conditional probability, Bayes rule, independence, and sampling.	B	C
Statistics and estimation	Likelihood, maximum likelihood, bias-variance trade-off, confidence intervals, bootstrap, and basic hypothesis testing.	B	C
Information theory	Entropy, cross-entropy, KL divergence, mutual information, and their use in learning objectives.	B	C
Optimization	Gradient descent, stochastic and mini-batch optimization, momentum, Adam/AdamW, learning-rate schedules, constrained optimization intuition, and convergence diagnostics.	B	C
Numerical computation	Floating-point behavior, numerical stability, vectorization, memory use, computational complexity, and stable implementations of common operations.	B	C
Discrete mathematics and graphs	Sets, relations, combinatorics, recursion, dynamic programming, graph terminology, paths, connectivity, and adjacency representations.	B	C
Advanced statistical foundations	Bayesian estimation, posterior prediction, multiple testing, and uncertainty quantification at a conceptual and applied level.	B	E

4. Programming, Data Handling, and Experimentation

Topic	Expected competency	Cat.	Lvl.
Python programming	Control flow, functions, modules, classes, comprehensions, iterators, exceptions, file handling, and standard data structures.	P	C
NumPy and tensor manipulation	Indexing, broadcasting, reshaping, masking, vectorized computation, reductions, and multi-dimensional array operations.	P	C
Pandas and structured data	DataFrames, joins, grouping, aggregation, missing values, categorical variables, time indices, and efficient input/output.	P	C
Visualization	Use Matplotlib and Seaborn to inspect distributions, learning curves, errors, embeddings, predictions, and model behavior.	P	C
Scikit-learn and boosting libraries	Build pipelines and use scikit-learn, XGBoost, LightGBM, or CatBoost appropriately for classical ML tasks.	P	C
PyTorch fundamentals	Tensors, datasets and dataloaders, modules, losses, optimizers, device placement, training loops, evaluation mode, and checkpointing.	P	C
CPU/GPU execution	Move data and models correctly, manage device memory, choose batch sizes, use mixed precision when appropriate, and diagnose bottlenecks.	P	C
Data processing	Normalization, standardization, imputation, padding, masking, tokenization, patching, resampling, augmentation, and variable-length data handling.	P	C
Feature engineering	Create compact informative features from tabular, categorical, image, text, audio, graph, and time-series data.	B	C
Validation and experimental design	Train/validation/test splitting, cross-validation, grouped and temporal splits, ablation studies, and prevention of data leakage.	B	C
Metrics and error analysis	Select, implement, and interpret task-appropriate metrics; inspect confusion matrices, ROC/PR curves, calibration, and subgroup performance.	B	C
Hyperparameter optimization	Manual search, grid/random search, efficient search strategies, early stopping, and fair model comparison.	P	C
Reproducibility and code quality	Random seeds, deterministic behavior, configuration management, logging, assertions, tests, clean notebooks, and reusable functions.	P	C
Efficient data pipelines	Caching, batching, parallel loading, memory mapping, sparse data, and profiling time and memory usage.	P	E

5. Artificial Intelligence Foundations

Topic	Expected competency	Cat.	Lvl.
Intelligent agents	Agents, environments, observations, actions, goals, utility, rationality, and the perception-action loop.	T	C

Uninformed search	Breadth-first, depth-first, uniform-cost, iterative deepening, completeness, optimality, and complexity.	B	C
Informed and heuristic search	Greedy best-first search, A*, admissibility, consistency, heuristic design, and beam search.	B	C
Adversarial search	Minimax, alpha-beta pruning, evaluation functions, stochastic games, and search-depth trade-offs.	B	C
Constraint satisfaction	Variables, domains, constraints, backtracking, propagation, ordering heuristics, and local search.	B	C
Planning	State-space planning, action models, goals, planning graphs, and the relation between planning and search.	B	E
Knowledge representation and logic	Propositional and first-order logic, inference, rule systems, ontologies, and limitations of symbolic representations.	T	E
Probabilistic reasoning	Bayesian networks, conditional independence, exact and approximate inference, hidden Markov models, and sequential belief updates.	B	E
Decision theory	Expected utility, risk, value of information, and decisions under uncertainty.	T	E
Multi-agent systems and game theory	Cooperation, competition, equilibria, mechanism intuition, and decentralized decision making.	T	E

6. Classical Statistical Learning

Topic	Expected competency	Cat.	Lvl.
Problem formulation	Regression, classification, ranking, clustering, density estimation, anomaly detection, and structured prediction.	B	C
Linear regression	Least squares, feature transformations, residual analysis, regularization, and interpretation.	B	C
Logistic regression	Log-odds, cross-entropy, decision thresholds, multiclass extensions, and calibration.	B	C
Regularization	L1 and L2 penalties, sparsity, weight decay, model capacity, and the bias-variance trade-off.	B	C
K-nearest neighbors	Distance metrics, scaling, neighborhood size, classification/regression, and computational trade-offs.	B	C
Decision trees	Splitting criteria, pruning, depth control, missing values, and interpretability.	B	C
Ensembles	Bagging, random forests, gradient boosting, stacking, and when ensembles improve robustness.	B	C
Support vector machines and kernels	Margins, soft-margin SVMs, kernel functions, and scaling considerations.	B	C
Probabilistic classifiers	Naive Bayes, linear/quadratic discriminant analysis, posterior probabilities, and calibration.	B	E
Model evaluation	Accuracy, precision, recall, F1, log loss, ROC-AUC, PR-AUC, ranking metrics, regression errors, and task-specific metrics.	B	C
Class imbalance and cost-sensitive learning	Resampling, class weights, threshold selection, focal-style objectives, and asymmetric error costs.	B	C
Clustering	K-means, hierarchical clustering, DBSCAN, spectral clustering, assumptions, and cluster validation.	B	C
Dimensionality reduction	PCA, t-SNE, UMAP, feature selection, visualization, and limits of low-dimensional projections.	B	C
Semi-supervised and weak supervision	Pseudo-labeling, consistency ideas, noisy labels, and use of unlabeled data.	B	E
Anomaly and novelty detection	One-class methods, isolation-based methods, reconstruction error, and thresholding.	B	E

Domain shift and generalization	Covariate shift, label shift, domain adaptation, invariance, and robust validation.	B	E

7. Neural Networks and Deep Learning

Topic	Expected competency	Cat.	Lvl.
Perceptrons and multilayer networks	Perceptron basics, multilayer perceptrons, universal approximation intuition, and network capacity.	B	C
Activations and losses	ReLU, sigmoid, tanh, softmax, MSE, MAE, cross-entropy, margin losses, and task-appropriate objectives.	B	C
Backpropagation	Forward and backward passes, computational graphs, gradient flow, and common implementation errors.	B	C
Optimization for deep learning	SGD, mini-batches, momentum, Adam/AdamW, learning rates, warmup, schedules, clipping, and convergence behavior.	B	C
Initialization and normalization	Weight initialization, batch normalization, layer normalization, and their effects on training.	B	C
Regularization	Dropout, early stopping, weight decay, augmentation, label smoothing, and model selection.	B	C
Embeddings and pooling	Learned representations for text, images, audio, categories, and graphs; max, average, and attention pooling.	B	C
Convolutional networks	Convolution, stride, padding, receptive fields, pooling, residual connections, and image/time-series use.	B	C
Recurrent sequence models	RNNs, LSTMs, GRUs, hidden state, sequence batching, teacher forcing, and vanishing/exploding gradients.	B	C
Attention and transformers	Attention mechanisms, self-attention, encoder/decoder blocks, masking, and positional information.	B	C
Autoencoders	Undercomplete, denoising, and variational autoencoders for representation learning and generation.	B	C
Transfer learning and fine-tuning	Pretrained encoders, feature extraction, full fine-tuning, parameter-efficient fine-tuning, and catastrophic forgetting.	B	C
Self-supervised and contrastive learning	Pretext tasks, augmentations, positive/negative pairs, contrastive objectives, and representation evaluation.	B	E
Training diagnosis	Learning curves, gradient checks, dead activations, unstable loss, overfitting, underfitting, and systematic ablations.	P	C

8. Reinforcement Learning and Sequential Decision Making

Topic	Expected competency	Cat.	Lvl.
Multi-armed bandits	Exploration-exploitation trade-off, epsilon-greedy methods, upper-confidence ideas, and regret intuition.	B	C
Markov decision processes	States, actions, transitions, rewards, policies, returns, discounting, and episodic/continuing tasks.	B	C
Value functions and Bellman equations	State/action values, Bellman expectation and optimality equations, and fixed-point intuition.	B	C
Dynamic programming	Policy evaluation, policy iteration, and value iteration when a model is known.	B	C
Monte Carlo and temporal-difference learning	Prediction from experience, bootstrapping, TD error, n-step returns, and eligibility-trace intuition.	B	C
Control methods	SARSA, Q-learning, on-policy vs off-policy learning, and exploration strategies.	B	C
Function approximation and deep RL	Value approximation, DQN, replay buffers, target networks, and instability sources.	B	E

Policy-gradient and actor-critic methods	REINFORCE, baselines, advantages, actor-critic structure, entropy regularization, and PPO.	B	E
Reward design and curriculum learning	Sparse rewards, shaping, legality/constraint rewards, staged objectives, and unintended incentives.	B	E
Imitation, offline, and model-based RL	Behavior cloning, offline data limitations, learned dynamics, planning with models, and sim-to-real intuition.	B	E
RL evaluation and reproducibility	Multiple seeds, return distributions, sample efficiency, stability, ablations, and fair comparisons.	B	C
Multi-agent reinforcement learning	Cooperative and competitive settings, non-stationarity, centralized training, and decentralized execution.	B	E

9. Computer Vision

Topic	Expected competency	Cat.	Lvl.
Image fundamentals	Pixels, channels, color spaces, resizing, interpolation, normalization, and common image formats.	B	C
Convolutional layers	Kernels, feature maps, receptive fields, padding, stride, pooling, and residual architectures.	B	C
Image classification	Training classifiers, transfer learning, pretrained encoders such as ResNet, and error analysis.	P	C
Object detection	Bounding boxes, IoU, non-maximum suppression, and practical use of YOLO, SSD, and DETR-style models.	B	C
Image segmentation	Semantic/instance segmentation, pixel-wise losses, U-Net-style models, and overlap metrics.	B	C
Image augmentation	Cropping, flipping, geometric and photometric transforms, mix-based augmentations, and label consistency.	P	C
Vision-language encoders	Joint image-text embeddings, CLIP-style similarity, zero-shot classification, and retrieval.	B	C
Generative vision	GANs, variational methods, diffusion models, conditioning, sampling, and evaluation limitations.	B	C
Video and temporal vision	Frame sampling, temporal pooling, motion features, sequence models, and video classification.	B	E

10. Natural Language Processing, Speech, and Audio

Topic	Expected competency	Cat.	Lvl.
Text preprocessing and tokenization	Normalization, sentence/word/subword tokenization, vocabularies, padding, masking, and sequence length.	B	C
Text representations	Bag-of-words, TF-IDF, static embeddings, contextual embeddings, and similarity measures.	B	C
Text classification and sequence labeling	Document classification, token classification, class imbalance, and evaluation.	P	C
Pretrained text encoders	BERT-style models, masked-language-model representations, pooling, and fine-tuning.	B	C
Language modeling	Autoregressive and masked objectives, likelihood, perplexity, generation, and decoding.	B	C
Encoder-decoder models	Sequence-to-sequence learning, attention, machine translation, summarization, and multimodal generation.	B	C
Retrieval and semantic search	Dense and sparse retrieval, embeddings, similarity, reranking, and retrieval evaluation.	B	E
Audio and signal representations	Waveforms, sampling rate, resampling, framing, spectrograms, Fourier transforms, and normalization.	B	C

Speech and audio models	Practical use of models such as Whisper, Qwen-Audio, and Voxtral when provided in the contest environment.	P	C

11. Transformers, Foundation Models, and Large Language Models

Topic	Expected competency	Cat.	Lvl.
Transformer mechanics	Scaled dot-product attention, multi-head attention, masking, positional encoding, residual connections, and normalization.	B	C
Architecture families	Encoder-only, decoder-only, encoder-decoder, mixture-of-experts intuition, and modality-specific transformers.	T	C
Tokenization and context	Subword tokenization, context windows, truncation, padding, attention masks, and prompt construction.	B	C
Pretraining objectives	Next-token prediction, masked modeling, denoising, instruction tuning, and their behavioral consequences.	T	C
Prompt engineering	Zero-shot/few-shot prompting, role and instruction design, examples, decomposition, context selection, and prompt evaluation.	B	C
Structured generation and decoding	Greedy, beam, sampling, temperature/top-p, constrained outputs, validation, repair, and stopping criteria.	B	C
Fine-tuning and PEFT	Supervised fine-tuning, LoRA/QLoRA, adapters, prompt tuning, dataset construction, and hyperparameter sensitivity.	B	C
Preference and reinforcement-based adaptation	Reward models, RLHF, DPO-style objectives, PPO/GRPO-style optimization, and alignment trade-offs.	B	E
Retrieval-augmented generation	Chunking, embeddings, vector search, retrieval, reranking, grounding, and citation-aware evaluation.	B	E
Tool use and agents	Function/tool calling, planning loops, memory, state tracking, environment feedback, and rule-constrained actions.	B	E
LLM evaluation	Task accuracy, exact match, validity, calibration, hallucination, robustness, preference evaluation, and contamination risks.	B	C
Inference efficiency	Quantization, batching, caching, sequence length, memory/latency trade-offs, and small-model selection.	B	E
Multimodal foundation models	Joint text-image-audio inputs, modality encoders, fusion, prompting, and cross-modal evaluation.	B	E
LLM safety and security	Prompt injection, jailbreaks, data leakage, unsafe outputs, tool misuse, and defensive validation.	B	C
Use of APIs and provided models	Understand open-source and API-based model workflows; actual contest access is governed by the Technical Appendix and task statement.	P	C

12. Time-Series, Sequence, and Sensor Learning

Topic	Expected competency	Cat.	Lvl.
Sampling and segmentation	Sampling frequency, windows, overlap, resampling, synchronization, variable-length sequences, and event segmentation.	B	C
Preprocessing	Filtering, detrending, normalization, missing data, outliers, padding, masking, and leakage-safe transformations.	B	C
Statistical and frequency-domain features	Moments, extrema, autocorrelation, spectral energy, dominant frequencies, and Fourier-based features.	B	C
Classical forecasting	Autoregression, moving averages, seasonality, trend, baseline forecasts, and ARIMA-style intuition.	B	E
Neural sequence models	RNN/LSTM/GRU, temporal CNNs, transformers, sequence-to-one, sequence-to-sequence, and multivariate inputs.	B	C

Multi-step forecasting	Direct and autoregressive prediction, teacher forcing, rollout, exposure bias, error accumulation, and horizon selection.	B	C
Temporal validation and metrics	Chronological splits, rolling evaluation, event-based splits, RMSE/MAE, scale-normalized metrics, and leakage control.	B	C
Anomaly and change-point detection	Point/contextual anomalies, reconstruction/prediction errors, change points, and threshold selection.	B	E
Sequence similarity and alignment	Cross-correlation, dynamic time warping, learned similarity, and invariance to phase or speed.	B	E
Sensor and IMU learning	Accelerometer/gyroscope data, axes, orientation effects, sensor fusion, activity recognition, and biometric verification.	B	E
Curriculum and long-horizon training	Progressively increasing sequence difficulty or forecast horizon while monitoring forgetting and stability.	B	E

13. Graph Machine Learning and Geometric Deep Learning

Topic	Expected competency	Cat.	Lvl.
Graph representations	Nodes, edges, attributes, directed/undirected graphs, adjacency matrices, edge lists, sparse tensors, and batching.	B	C
Classical graph features	Degree, centrality, neighborhoods, paths, connected components, Laplacian intuition, and handcrafted graph statistics.	B	E
Message passing	Neighborhood aggregation, update functions, receptive fields, permutation invariance, and oversmoothing intuition.	B	C
GNN architectures	GCN, GraphSAGE, GAT, edge-aware message passing, residual connections, and normalization.	B	C
Graph learning tasks	Node classification/regression, link prediction, edge prediction, graph classification/regression, and graph embeddings.	B	C
Pooling and readout	Node-to-graph aggregation, hierarchical pooling intuition, and global representations.	B	E
Graph construction	Building edges from physical relations, similarity, proximity, domain knowledge, and learned connectivity.	B	C
Temporal and spatio-temporal graphs	Dynamic node features, temporal message passing, recurrent/temporal GNNs, and spatial-temporal forecasting.	B	E
Heterogeneous and dynamic graphs	Multiple node/edge types, time-varying topology, relation-specific parameters, and practical representations.	B	E
Scalability and implementation	Sparse operations, neighbor sampling, mini-batching, memory constraints, and graph data loaders.	P	E
Domain and physical constraints	Directional flow, conservation, topology-aware losses, and integrating scientific priors into graph models.	B	E

14. Assessment and Task-Setting Expectations

Tasks may require writing code, fitting models, running inference, producing predictions, analyzing results, or combining these activities.

A task may span multiple data modalities, including tabular data, images, text, audio, video, time series, sensor streams, and graphs.

Core topics may be assessed directly. Extended topics should be supported by an at-home task, a concise tutorial, starter code, or sufficient task-specific explanation.

Contestants should not be expected to memorize obscure library APIs; task design should reward reasoning, modeling, data handling, and experimentation.

All required checkpoints, datasets, documentation, and software should be available within the official contest environment.

Tasks should permit meaningful partial solutions, including simple baselines, feature-based methods, classical ML, and progressively stronger models where appropriate.

Evaluation should use clearly defined metrics, robust hidden test data, and splits that prevent identity, temporal, group, or near-duplicate leakage.

Environment alignment
The current ICAIC rules identify Python and Jupyter as the primary contest environment, with PyTorch and scikit-learn as core libraries, identical local compute, and limited internet access. The final public syllabus should remain consistent with the annual Technical Appendix. [6]

15. Reference Basis and Recommended Resources

These resources guide scope and preparation. They are not intended to make every chapter examinable; the topic tables above are normative.

Resource	Intended role
IOAI 2026 Syllabus	Normative base coverage and the Theory/Practice/Both classification.
Artificial Intelligence: A Modern Approach - Stuart Russell and Peter Norvig	Classical AI, search, reasoning, planning, probabilistic decision making, agents, and responsible AI.
The Elements of Statistical Learning - Trevor Hastie, Robert Tibshirani, and Jerome Friedman	Statistical learning, regularization, model selection, kernels, trees, ensembles, and unsupervised methods.
Reinforcement Learning: An Introduction - Richard S. Sutton and Andrew G. Barto	Sequential decision making, value-based methods, policy methods, and exploration.
Deep Learning - Ian Goodfellow, Yoshua Bengio, and Aaron Courville	Neural networks, optimization, regularization, convolutional and sequence models, and representation learning.
Modern transformer and foundation-model literature/documentation	Attention, transformers, LLM training and adaptation, retrieval, agents, multimodality, evaluation, and safety.

Supplementary resources

Bishop, C. M., & Bishop, H. (2024). *Deep learning: Foundations and concepts*. Springer. https://doi.org/10.1007/978-3-031-45468-4

Official companion website: https://www.bishopbook.com

Murphy, K. P. (2022). *Probabilistic machine learning: An introduction*. MIT Press. https://probml.github.io/pml-book/book1.html

Publisher page: https://mitpress.mit.edu/9780262046824/probabilistic-machine-learning

Murphy, K. P. (2023). *Probabilistic machine learning: Advanced topics*. MIT Press. https://probml.github.io/pml-book/book2.html

Publisher page: https://mitpress.mit.edu/9780262048439/probabilistic-machine-learning

For the latest third-edition online manuscript:

Jurafsky, D., & Martin, J. H. (2026). *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition with language models* (3rd ed.; online manuscript released January 6, 2026). https://web.stanford.edu/~jurafsky/slp3

For the published second edition:

Jurafsky, D., & Martin, J. H. (2009). *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition* (2nd ed.). Pearson Prentice Hall. ISBN: 978-0-13-187321-6.

Hamilton, W. L. (2020). *Graph representation learning.* Springer.
https://doi.org/10.1007/978-3-031-01588-5

Author's book website: https://www.cs.mcgill.ca/~wlh/grl_book

Author-provided PDF: https://www.cs.mcgill.ca/~wlh/grl_book/files/GRL_Book.pdf

(version number to be updated)

PyTorch https://docs.pytorch.org/docs/stable

Scikit-learn https://scikit-learn.org/stable

NumPy https://numpy.org/doc/stable

Pandas https://pandas.pydata.org/docs

Hugging Face https://huggingface.co/docs