OBJECTIVE 2 – QUALIFICATION BENCHMARKING

Bachelor in Computer Science
Ho Chi Minh City University of Technology (HCMUT)

Table of Contents

Introduction

The Bachelor in Computer Science at Ho Chi Minh City University of Technology provides fundamental concepts and applications of computer science and how they can be applied to business, finance and economics. This is a multidisciplinary programme and for this benchmarking exercise we focus only on modules/courses related to computer science.

For this benchmarking exercise we have developed a scoring matrix where we identified 5 themes (programming, knowledge management, knowledge abstraction, knowledge representation/communication and research/soft skills). **Programming** theme entails criteria related to design and development of not only software, but also other artefacts like algorithms, network, IoT framework etc. The theme also includes the evaluation process and collaborative management of the artefacts. **Knowledge management** primarily focuses on processes and techniques of warehousing different types of data. The theme also includes security and privacy issues related to data management. **Knowledge abstraction** theme focuses on different data analytics and machine learning techniques applied to different types of data. **Knowledge representation/communication** theme includes different visualisation techniques used to represent the results (from database query through to data analytics to algorithm) to a wide range of stakeholders. **Research/Soft skills** theme focuses on the understanding and practice of research methods along with the ability to undertake teamwork and present results to a wider audience.

Within each theme, we have a set list of criteria against which each course is scored. The score is within the range of 50 – 100. 90 – 100 (fully meets the criteria); 75 – 89 (mostly meets the criteria); 60 – 74 (partially meets the criteria); 50 – 59 (barely meets the criteria). The marks are indeed subjective and therefore debatable. However, the pattern that emerges as a result of the scoring of each module/course provides a holistic view of the programme and clearly identifies the areas of strengths and improvements.

Design of the programme

1. This 128 credit programme equips students with the capability to address different areas of computer science with the options of majoring in Computer Science, Applied Artificial Intelligence, Cybersecurity, Data, Security Engineering and Business Intelligence Data Engineering, Image Processing and Computer Vision by taking appropriate elective courses in the final two years.
2. The programme develops a sequential accumulation of knowledge and expertise. Starting with compulsory courses (78 credits) providing the foundational knowledge in mathematics, statistics, basic science, law, philosophy, English as well as an introduction to computer engineering and programming.
3. The elective courses are divided into four groups; group A provides experience in different types of programming projects, while group B in multidisciplinary projects.
4. Group C is elective courses that help to shape specialisation. Group D provides elective courses in economics and management.
5. Finally, the speciality courses reinforce the specialisation pathway along with internship and thesis projects to gain applied and research experience in the specialisation.
6. For this benchmarking exercise only data/computer science related courses were evaluated.

Mode of Delivery

7. All courses are delivered in English and have theory, practice and self-study credit hours. From the course structure document, we could not identify any course with allocated self-directed study time.
8. Most of the courses are theory based and as such assessments are exam/class test based.
9. The programme provides a strong emphasis on soft skill development with emphasis on project management.
10. From the documents provided, it is not clear how the theory part of each course is delivered. A strong foundation in maths and the development of soft skills creates a multidisciplinary mindset for the students, which in the long run helps them to understand the underpinning mathematical basis of advanced subjects like Cryptography.
11. For practical sessions, information regarding class size and available resources is necessary to evaluate the effectiveness of the practical sessions.

Learning and Teaching

12. The programme offers a wide range of elective courses in different topics for specialisation. Depending on the area of specialisation the number of courses offered varies (~ 35 courses). Delivering such a wide range of courses requires resources and may impact course quality. Therefore, combining some courses would benefit the students as well as staff, in terms of resources and quality.
13. The programme provides an excellent foundation in mathematics (algebra etc.) and statistics. This is critical for all computer science related courses. However, due to the wide variety of programming related courses (software, web, game, mobile programming) the students' choice may get too diversified, leading to lack of depth in programming. Focusing two or three courses in object oriented (java) and scripting (python) from a traditional software engineering and data science point of view would benefit students to develop a strong foundation in programming.
14. The programme has a wide range of project management skill development courses including multidisciplinary project management courses along with courses in economics and business administration. This structure enables students to develop leadership roles in industries and apply their technical skills through a multidisciplinary team.
15. The programme lacks courses on cloud-based unstructured/real-time data warehousing systems like AWS/Datalake.
16. Critical understanding about code sharing (github etc.) seems to be missing. For data science programmes it is important that students are aware and have experience of code sharing, documentation and different licenses used for open-source software/algorithm development.

Assessment and feedback

17. No information was available with regard to assessment or exam samples.
18. No information was provided as to how student feedback is captured, evaluated and utilised for the improvement of the courses.

Conclusion and Recommendations

19. Teaching modality

a. More discussion-based teaching approaches including flip classroom type teaching model can be introduced to increase student engagement and self-directed study.
b. Industry 4.0 focused case studies can be implemented to get more knowledge about different real-life projects, their shortcomings etc.

20. Teaching content

c. Low code/No code-based programming are becoming popular (10.3390/electronics10101192) in universities with the rise of online education and as a result of COVID-19. Adaptation with new trends will help students to develop new applications/algorithms more easily. This impacts not only skill development but confidence also.
d. More emphasis can be given to scripting languages like Python as well as users' requirement analysis.
e. In this regard API based programming like ChatGPT (from OpenAI etc.) to any software/app would benefit students with high quality trained dataset/model integration.
f. Analysis of real-life data from different domains (finance, healthcare, social media etc.) is essential to get an understanding of different design, development and deployment of IT in these domains.
g. Engagement with stakeholders and requirement capture is pivotal. Therefore, with different types of programming/machine learning courses these aspects need to be included.
h. Use of online content/courses can introduce students to new topics and a choice of learning sources (in contrast to recommended books). This diversity of content and modality of delivery not only helps students to be in line with current trends but also initiates peer learning.
i. Skill development on code sharing (through github etc.) and open licence needs to added to the course curriculum along with collaborative code development (e.g. Google Colab, AWS).
j. Basic understanding of how to protect intellectual property rights related to algorithms and the process of protecting these rights through third party.
k. Critical understanding of research methods in higher education and steps involved from idea generation through to publication and/or application can be incorporated.
l. Basic knowledge of social media-based profile creation e.g. LinkedIn profile that will facilitate future job prospects can be incorporated to develop profile.

21. Assessment

m. More emphasis on project-based assessments (instead of exams) would help students to get experience of teamwork and other aspects of project management.

CRITERIA	Calculus 1	Calculus 2	Algebra	Probability and Statistics	Discrete structure for computing	Mathematical Modeling	Introduction to Computing	Digital Systems	Programming Fundamentals	Data Structures and Algorithms	Computer Architecture	Database Systems	Advanced Programming	Introduction to Computer Programming	Software Testing	Software Architecture	Advanced Software Engineering	Systems Analysis and Design	Database Systems	Distributed Calculations	Data Mining	Algorithm Design	Information Security	IoT and App Dev	Intelligent System	Mobile App Dev	Adv. Comp. Network	Web development	Applied Cryptography	Business Information Systems and Processes	Natural Language Processing
PROGRAMMING																															
General knowledge on different software/algorithm, their application, strenghts and limitations.							98	96	93	82			97	94	97		93	98		96			81	81							
Knowledge and skill to undertake requirement analysis. Ability to collect and summarise the requirements. Also understand the market demand/trend in context of software/algorithm development.							93										91	84					93	72				91			
Ability to work in a team environment and understanding of the importance of communication in a multi disciplinary/institutional team environment.							87										94						92			65					
Demonstrate the knowledge and skill of project management. Particularly the differentiation and applicability of different types of project management approaches and the relevance to the project in hand.																										87					
Demonstrate critical understanding of the software/algorithm design approaches and availability of design tools							92	93					91	94	92			92		89						93					
Demonstrate critical understanding of the software/algorithm development approaches and availability of development tools/languages.							82	89					93	92																	
Critical knowledge about the importance of software/algorithm evaluation/testing and the steps of releasing.							83							91	85		81														
Critical knowledge about the importance of software/algorithm security related issues.																															
Demonstrate critical knowledge on both object oriented and scripting languages.							81	83					89	96	83																
Ability to code in distributed cloud as well as stand-alone environement using wide range of programming languages.													84							96								72			
Good knowledge on privacy, security issues related to software/hardware/algorithm.																							96				84				
Critical understanding of code sharing, archiving and protection through cloud-based repository platforms.																															
Good knowledge about different cloud based computing platforms, e.g. AWS																															
Understanding about opensource/stand-alone/cloud-based software/hardware/algorithm.														84																	
Critical knowledge on computing performance and how to address performance related challenges.														82																	
KNOWLEDGE MANAGEMENT																															
Good understanding about big data and their impact on business and society.																															
Critical knowledge about different steps of data management pipeline – from collection to data analytics												81							81												
Critical understanding about different data types, sources of data types and their strengths and limitations.												87							87												
Critical understanding about different data collection processes and resources along with regulations associated to the processes including ethics and permission requirements.												82							82												
Demonstrate the knowledge on data privacy and it differs in different domains – manufacturing, medical etc.																															
Critical understanding about data quality, particularly standard process of measuring data quality.												82							82												
Demonstrate the appreciation towards the importance of data cleaning and knowledge on different steps, tools used for data cleaning.																															
Understanding about the data validation process and its importance towards data management.												94							94								86				
Good knowledge about different data warehousing techniques and technical details for implementing different types of databases.												91							91												
Updated knowledge about data warehousing for both structured and unstructured data.												71							71												
Critical knowledge about the approach to query the data as per the requirements of the users.												95							95	93											
Good understanding about the process of data transformation in relation to the users requirements.																															
Provide detailed knowledge and skill on different data security related issues and how to identify different types of threats.																															
Explain different types of data protection measures that can be taken to safeguard data breach.																															
Understand institutional policies related to data security and privacy and tools required to implement these policies.												93							93	89											
Good understanding about intellectual property rights and governance laws.																															
KNOWLEDGE ABSTRACTION																															
Demonstrate good knowledge of algebra and calculus that enables students to refresh their high school math knowledge.	97	94	91			98				72												95			95						
Provide detailed understanding of distribution and probability statistics.				94	91																										
Critical understanding on intelligent agent and different types of logic representations.			65							92											83								92		92
Different algorithms (structured, unstructured, adaptive) – their applications and refinement of algorithm parameters.									95	91											74								83		87
Critical knowledge on algorithm testing, evaluation and optimisation.									86	83											81								65		73
Good understanding of ethics and policy related to algorithm development and deployment.																															
Up to date knowledge on state of the art algorithms implemented by different stakeholders.										78																					
Basic understanding on how to protect intellectual property rights related to algorithms and the process of protecting these rights through third party.																															
Provide detailed understanding of algorithms used for image analysis and object detection from including timelapse and multimodal images.																															
KNOWLEDGE REPRESENTATION/COMMUNICATION																															
Evaluate and apply data visualisation grammar and principle to the whole of the visualisation process and the resulting presentations.																															
Evaluate the capabilities of different visualisation tools and programming languages, both proprietary and open-source, to support the discovery and display of critical and valuable answers hidden in small, medium and large data.																															
Implement analysis and visualisation techniques using realistic data sources from disparate disciplines and using the most appropriate visualisation tools in order to identify the valuable questions and to develop well justified, actionable answers.																															
Good understanding of processes to know the audience in terms of their requirements, domain knowledge etc.																															
Differentiate between exploratory and explanatory data visualisation and or knowledge representation.																															
RESEARCH/SOFT SKILLS																															
Understanding of core aspects of philosophy and its sub discipline.														72									84							86	
Explore and express different ideas of philosophy and their relevance to research in higher education.																															
Introduction to the main ideas of logic and methods of recognising relevant information to construct persuasive arguments.																														83	
Develop analytical, argumentative problem-solving skills.														76									81							89	
Develop critical thinking and problem-solving skill in a collaborative manner.														78									93							94	
Critical understanding of research methods in higher education and steps involved from idea generation through to publication and or application.																							83								
Critical understanding of project management and role of different team members at different levels along with the interplay.																														83	
Knowledge about different administrative and operational tasks required for successful project management.																														86	
Basic understanding of activities related to dissemination of research outcome to wider audience.																														81	
Basic knowledge of social media-based profile creation e.g. LinkedIn profile that will facilitate future job prospects.																															