EQUIP
UK-VN
Enabling an environment of QUality for International HE Partnerships between the UK and Viet Nam

BRITISH COUNCIL

OBJECTIVE 2 – QUALIFICATION BENCHMARKING

Bachelor of Data Science
Hanoi University of Science and Technology (HUST)

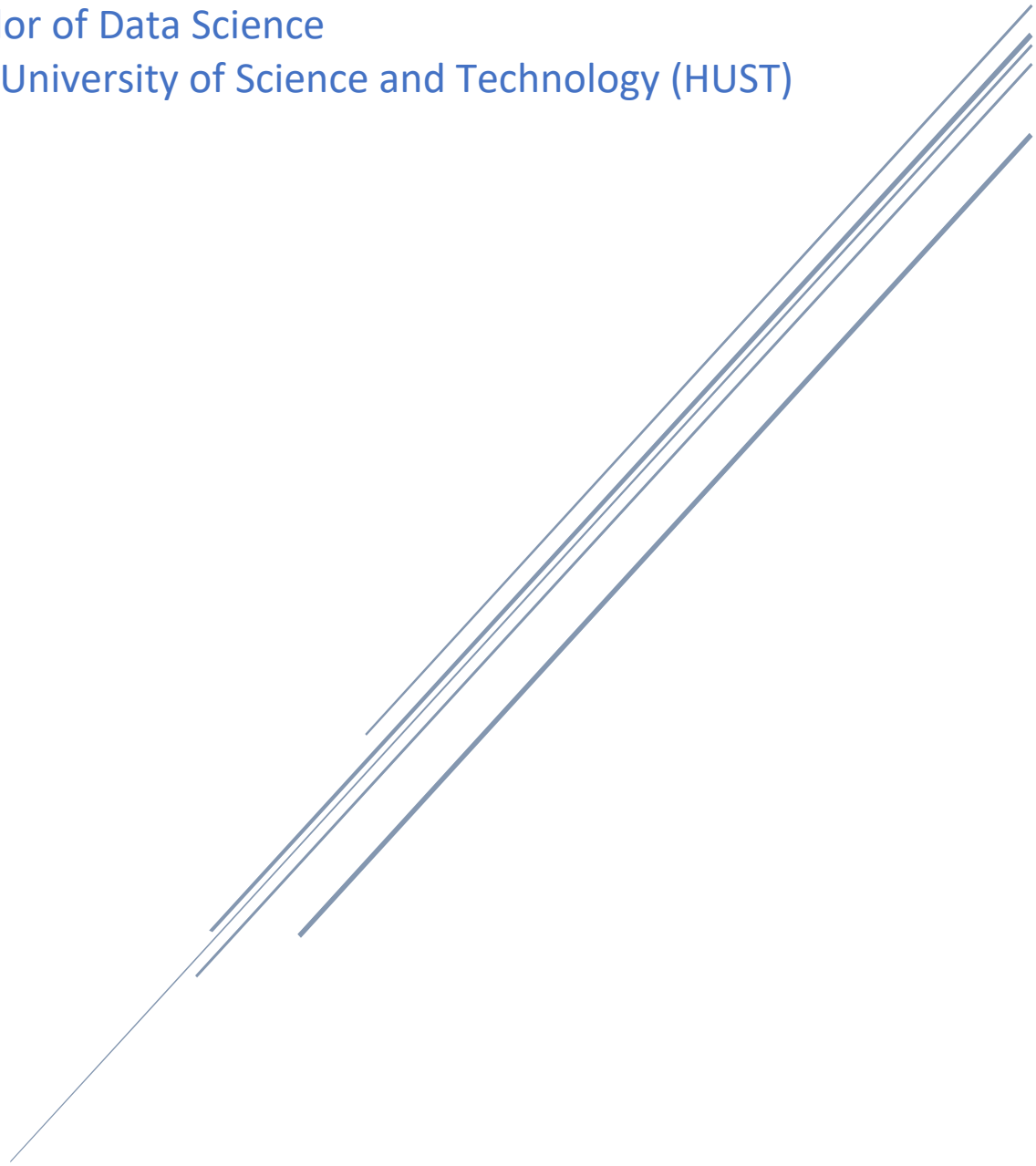

Table of Contents

Introduction

The BSc Data Science (DS) programme at Hanoi University of Science and Technology (HUST) addresses the needs of local employers and addresses the challenges of the digital transformation ambition of the Vietnamese industries and government. The programme provides a blended knowledge of data science along its application and professional management and leadership skill development.

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 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. The programme is integrated with a master's programme. For this benchmarking exercise the focus is only on the UG part. The BSc programme progresses with different types of knowledge and skill development modules. It starts with offering basic knowledge of philosophy, politics and law along with courses on physical education and English (19 credits), followed by courses on maths and science (32 credits) providing foundational knowledge. The programme then focuses on computer engineering courses (49 credits) followed by data science specialisation courses (15 credits) and soft skill development courses (9 credits). The programme ends with a thesis (8 credits) submission.
2. Credit for general courses ranges between 2 to 4, while most of the compulsory courses are 3 credits, the elective courses are 2 credits.

Mode of delivery

3. All courses are delivered in English and have theory, practice and self-study credit hours. The self-study hours vary with students and are usually 2 hours/credit/week.
4. The programme has six elective management/entrepreneurship focused courses. Acquiring such management skills is pivotal in the ICT sector. From the syllabus it is evident that most of the courses are exam based and courses have projects through

which students experience teamwork. Setting marks (10-20%) to these informal projects would formalise the work/collaboration undertaken in these projects.

5. The courses are delivered through a mixture of lectures, discussions and case studies. To further increase student engagements, flipped classroom or peer assessment approaches can be introduced.
6. For practical sessions, information regarding class size and available resources is necessary to evaluate the effectiveness of the practical sessions.

Learning and Teaching

7. With regard to the programming theme, most of the courses addressed different aspects of software and/or algorithm design and development. The predominant programming language was not evident from the syllabus (Python is covered only in one course and Java in the other).
8. Within the programming theme, the security aspect of software/algorithm design and development can be improved by incorporating some topics from cyber security particularly access control to source code and sensitive data (although there is a course on Cryptography and security).
9. Critical understanding about code sharing (github etc.) seems to be missing. For modern ICT 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.
10. In addition to textbooks (required/recommended), online materials could be included. Books require updated versions and need change (one of the required books is a 1997 edition).
11. Communicating the results and/or artifacts are an important part of any ICT programme. Including a presentation part with most of the model enables students to develop their communication skills. The programme in this regard has two visualisation courses with focus on business data. Including other data like scientific experimental data, real time data would benefit the visualisation based courses.
12. Courses focusing on algorithms (e.g. Introduction to Deep learning) can be improved by reducing the amount of theoretical knowledge and incorporating more case studies and/or application of the algorithm in real life scenarios. In these approaches students would gain more experience of the application of the appropriate algorithm in the right context.

Assessment and feedback

13. No information was available about assessment or exam samples.
14. No information was available about student feedback mechanism and evaluation.

Conclusion and Recommendations

15. 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. Project-based learning approaches can be implemented to get more knowledge about different real-life projects, their short comings etc.
16. Teaching content
    a. 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.

b. In this regard API based programming like GPT-3 like language model (from OpenAI etc.) to any software/app would benefit students with high-quality trained dataset/model integration.
c. Analysis of real-life data from different domains (finance, healthcare, social media etc.) is essential to get an understanding about different data sources and types.
d. Engagement with stakeholders and requirement capture is pivotal. Therefore, with different types of programming/machine learning courses these aspects need to be included.
e. Use of online content/courses can introduce students to new topics and choice of learning sources (in contrast to the recommended books). This diversity of content and modality of delivery not only helps students to be in line with current trends, but also initiate peer learning.
f. Skills development on code sharing (through github etc.) and open licence needs to be added to the course curriculum along with collaborative code development (e.g. Google Colab, AWS).

17. Assessment
    a. More emphasis on project-based assessments (instead of exams) would help students to get experience of team work and other aspects of project management.

CRITERIA	MATH & BASIC SCIENCE (32)										MAJOR COURSES (49)																	SOCIAL COURSES (9)						CONCENTRATION (15)				
	Calculus I	Calculus II	Calculus III	Algebra	Basic physics	Introduction to Programming	Probability and Statistics I	Discrete Mathematics	Fundamentals of optimization	Applied Statistics and Experimental Design	Data Structures and Algorithms	Computer Architecture	Database	Object-oriented Programming	Machine Learning	Mobile Programming	Introduction to Cryptography and Security	Operating Systems	Computer Networks	Systems Analysis and Design	System Programming	Introduction to Software Engineering	Introduction to Data Science	Introduction to Deep Learning	Natural Language Processing	Parallel and Distributed Programming	Data Visualization	Technical Writing and Presentation	Introduction to Management	Business Culture and Entrepreneurship	Applied Psychology	Technology and Technical Design Thinking	Industrial Design	Big Data Storage and Processing	Introduction to Business Analytics	Systems Analysis and Design	Web Mining	Mathematical Modelling
PROGRAMMING																																						
General knowledge on different software/algorithm, their application, strenghts and limitations.														73						98	95	93	84			67												
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.																		78		84	87	91	72															
Ability to work in a team environment and understanding of the importance of communication in a multi disciplinary/institutional team environment.																65						94				94												
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.														84		87																						
Demonstrate critical understanding of the software/algorithm design approaches and availability of design tools												92				93		87		92			86			87												
Demonstrate critical understanding of the software/algorithm development approaches and availability of development tools/languages.												84		93				76			67																	
Critical knowledge about the importance of software/algorithm evaluation/testing and the steps of releasing.												74		84								81	63															
Critical knowledge about the importance of software/algorithm security related issues.																			98																			
Demonstrate critical knowledge on both object oriented and scripting languages.																																						
Ability to code in distributed cloud as well as stand-alone environement using wide range of programming languages.																																						
Good knowledge on privacy, security issues related to software/hardware/algorithm.																			78																			
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.																																						
Critical knowledge on computing performance and how to address performance related challenges.																																						
KNOWLEDGE MANAGEMENT																																						
Good understanding about big data and their impact on business and society.													93																						89			
Critical knowledge about different steps of data management pipeline – from collection to data analytics													85																					72	83			
Critical understanding about different data types, sources of data types and their strengths and limitations.													78																					63				
Critical understanding about different data collection processes and resources along with regulations associated to the processes including ethics and permission requirements.																																						
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				
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.																																			57			
Good knowledge about different data warehousing techniques and technical details for implementing different types of databases.													75																						89			
Updated knowledge about data warehousing for both structured and unstructured data.																																						
Critical knowledge about the approach to query the data as per the requirements of the users.													72																									
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.																	92																					
Explain different types of data protection measures that can be taken to safeguard data breach.																	95																					
Understand institutional policies related to data security and privacy and tools required to implement these policies.																	63																					
Good understanding about intellectual property rights and governance laws.																	68																					
KNOWLEDGE ABSTRACTION																																						
Demonstrate good knowledge of algebra and calculus that enables students to refresh their high school math knowledge.	97	94	96	91																																		
Provide detailed understanding of distribution and probability statistics.					89		94	91		95																												
Critical understanding on intelligent agent and different types of logic representations.				65	76																			69													84	85
Different algorithms (un/supervised, adaptive) – their applications and refinement of algorithm parameters.						93			74		72				87									92	94												82	73
Critical knowledge on algorithm testing, evaluation and optimisation.						64			97		67				71									73	87													
Good understanding of ethics and policy related to algorithm development and deployment.																								62														
Up to date knowledge on state of the art algorithms implemented by different stakeholders.											67																											67
Basic understanding on how to protect intellectual property rights related to algorithms and the process of protecting these rights through third party.																																					63	
Provide detailed understanding of algorithms used for image analysis and object detection from including timelapse and multimodal images.															74																							
KNOWLEDGE REPRESENTATION/COMMUNICATION																																						
Evaluate and apply data visualisation grammar and principle to the whole of the visualisation process and the resulting presentations.																											91	67										
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.																											83											
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.																											71											
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.																												87	85	75		93	89		73			
Explore and express different ideas of philosophy and their relevance to research in higher education.																												82										
Introduction to the main ideas of logic and methods of recognising relevant information to construct persuasive arguments.																														89					93	81		
Develop analytical, argumentative problem-solving skills.																												91				86				91		
Develop critical thinking and problem-solving skill in a collaborative manner.																												78					75					
Critical understanding of research methods in higher education and steps involved from idea generation through to publication and or application.																													89							92		
Critical understanding of project management and role of different team members at different levels along with the interplay.																														91		74						
Knowledge about different administrative and operational tasks required for successful project management.																													96	94			72					
Basic understanding of activities related to dissemination of research outcome to wider audience.																												84										
Basic knowledge of social media-based profile creation e.g. LinkedIn profile that will facilitate future job prospects.																																						