Background Worksheet
Faculty of Computer Science – Dalhousie University

Name of applicant: Mohammedfaizal Gulammustufa Maulvi

Dalhousie Course Number	Equivalent Course Number(s)	Equivalent Course Name(s)	Year (1–5)	Calendar Year	Grade	University
CSCI 1110	2110003	Computer Programming and Utilization	1	Winter 2017	AB	Gujarat Technological University
CSCI 2110	2130702	Data Structure	2	Winter 2018	BC	Gujarat Technological University
CSCI 2122	2150708	System Programming	3	Winter 2019	AB	Gujarat Technological University
CSCI 2134	2160701	Software Engineering	3	Summer 2020	AA	Gujarat Technological University
CSCI 2141	2130703	Database Management Systems	2	Winter 2018	BB	Gujarat Technological University
CSCI 3110	2150703	Analysis and Design of Algorithms	3	Winter 2019	AB	Gujarat Technological University
CSCI 3120	2140702	Operating System	2	Summer 2019	BB	Gujarat Technological University
CSCI 3130	2160701	Software Engineering	3	Summer 2020	AA	Gujarat Technological University
CSCI 3136	2140705	Object Oriented Programming with C++	2	Summer 2019	BB	Gujarat Technological University
CSCI 3136	2140709	Computer Networks	2	Summer 2019	AB	Gujarat Technological University
	2150002	Cyber Security	3	Winter 2019	AA	Gujarat Technological University

Instructions

In the table, for each Dalhousie course in the first column, list up to two courses that you took and that you believe covered the same topics as this course. For each equivalent course you list course, provide the course number and course name under which it appears in your transcripts, the year of study and calendar year when you took the course, the grade you obtained, and the university at which you took the course. To help you assess which of the courses you took may be equivalent to the listed Dalhousie courses, a brief summary of each of the Dalhousie courses in the table is given below.

Course summaries

CSCI 1110: This course provides a general introduction to computer science and the hardware and software of computers. The main

focus is on programming skills and how to apply these skills in solving a variety of problems. Algorithmic concepts, linear data structures are emphasized.
NOTES: This course assumes that the student has some programming experience.

CSCI 2110: This course provides a comprehensive introduction to data structures and algorithms, including their design, analysis, and implementation. In
discussing design and analysis, there is a strong emphasis on abstraction. In discussing implementation, general approaches that are applicable in a wide range of procedural programming language are emphasized, in addition to a focus on the details of implementations.

CSCI 2122: This course presents tools, techniques, and concepts for systems programming. Students will be exposed to assembly and will be introduced to the C programming language. The course also discusses memory hierarchies, performance measurement, exception control flow, and performance related issues. Students will be involved in significant amount of low-level programming.

CSCI 2134: This course presents techniques and methodologies for software development with modern tools. It introduces students to the software development life cycle and best practices for source code management, testing, debugging, and building. Particular focus will be on building individual skills in the development and testing phases of the life cycle. Students will be expected to work with existing medium-size code-bases that are implemented in different programming languages. Students will be exposed to a variety of software tools and will be expected to use them throughout the course.
CSCI 2141: This course introduces students to the concepts of database management systems and database design. Topics include database (DB) components, DB design using entity relationships (relational and object-oriented), SQL, and transactional properties and techniques to support them. The concepts will be reinforced using one or more database management systems.

CSCI 3110: This course covers techniques for the design and analysis of efficient algorithms and data structures. Topics include asymptotic analysis, divide and conquer algorithms, greedy algorithms, dynamic programming, data structure design, optimization algorithms, and amortized analysis. The techniques are applied to problems such as sorting, searching, identifying graph structure, and manipulating sets.

CSCI 3120: This course includes a review of I/O and interrupt structures. Topics covered include dynamic procedure activation, system structure and evaluation, memory management, process management, process scheduling, recovery procedures, concurrency, deadlocks, resource allocation, protection, and operating systems implementation.

CSCI 3130: The course examines the process of software development, from initial planning through implementation and maintenance. A brief survey of available tools and techniques will be presented covering the topics of analysis, planning, estimating, project management, design, testing, and evaluation. Particular emphasis will be given to organizing and planning, team participation and management, top-down design and structure charts, system and information flow diagrams, walk-throughs and peer review, and testing and quality control.

CSCI 3136: This course provides a comparative study of advanced programming language features. Topics include statement types, data types, variable binding and parameter passing mechanisms. Formal methods for syntactic and semantic description of programming languages are examined.

CSCI 3171: This course gives students a foundation in computer networks. It presents a top-down view of the layered architectural elements of communication systems, focusing on the Internet and TCP/IP. Topics include client/server systems, packet switching, protocol stacks, queuing theory, application protocols, socket programming, remote service calls, reliable transport, UDP, TCP, and security.