Michael Fox

CS-157A

February 11, 2025

Hands On #1

Using the provided database schema, write relational algebra expressions to answer the following queries:

**Database Schema:**

1. **Students** (StudentID, Name, Age, Major)
2. **Courses** (CourseID, CourseName, Department)
3. **Enrollments** (StudentID, CourseID, Grade)
4. **Professors** (ProfessorID, Name, Department)
5. **Teaches** (ProfessorID, CourseID, Semester)

Queries:

1. Find all students who are majoring in 'Computer Science'. (use Selection)

R1 = Signma\_Major= ‘Computer Science’(Students)

1. List all distinct course names offered by the 'Mathematics' department. (use Projection)

R1 = Pi\_CourseName(Courses)

1. Find the names of all students and the courses they are enrolled in. (use Natural join)

R1 = Students *Join* Courses

1. Retrieve all students who have received a grade higher than 'B' in any course. (use Theta join)

R1 = Sigma\_Grade>’B’(Enrollemnts)

R2 = R1 *Theta Join*\_Enrollments.Studnent.ID = Students.StudentID (Students)

1. List all possible pairs of students and professors. (use Cartesian Product)

R1= Students X Professors

1. Find students who are enrolled in courses taught by Professor 'Smith' and also courses taught in the 'Fall' semester. (use Set Operations)

R1 = Pi\_StudentID (Pi\_CourseId(Sigma\_Name = ‘Smith’(Professors) *Join* Teachers) *Join* Enrollemnts)

R2 = Pi\_StudentId(Pi\_CourseId(Sigma\_Semester = ‘Fall’(Teachers)) *Join* Enrollments)

R3 = R1 *Intersection* R2

1. Calculate the total number of courses each student is enrolled in. (use Extended Projection)

R1 = StudentId, Count(CourseId) 🡪TotalCourses(Gamma\_StudentId,Count(CourseId)(Enrollements))

1. Rename the attributes of the **Professors** table to (ProfID, ProfName, Dept).

R1 = *rho \_*Profs(ProfID, ProfName, Dept.) (Professors)