**Course Scheduling Features**

Feature 1: new course

Input: course name, program id, number of credits, grading format, whether the course is required, room type, number of sections.

Output: if a course with the same name exists then update the existing course with the values or insert the new course.

Exec add\_course(‘Internetworking’, ‘IS 6900’, 3, ‘pass’, 1, ‘Class room’, 1);

Feature 2: new instructor

Input: instructor name, department id, and instructor type

Output: if an instructor with the same name exists at the same department with same instructor type then print an error message. Otherwise insert the instructor and generate instructor id.

Exec add\_instructor(‘Jay Patel’, ‘IS’, ‘part-time’);

Feature 3: teaching preferences

Input: instructor id, year, semester, course load, the courses the instructor is willing to teach, the number of sections the instructor is willing to teach, and up to two days of week that the instructor cannot teach.

Output: Please first check special cases including 1) the input instructor id is valid. 2) the list of courses is smaller than the course load, 3) the number of blackout days is over two. Print an error message if one of these cases occurs. Otherwise insert these input to appropriate tables.

Exec teaching\_preference(8, 2019, ‘second’, 2, ‘Telenetworking’, 1, 2);

Feature 4: allow students to search all courses for given semester

Input: year, semester, program id

Output: The procedure prints out for each scheduled course section the name of the course, number of credits, grading format, schedule id, section id, name of instructor, name of classroom, days of class, start and end time of each class, and whether the class is open or full. Please order the results by course id and section number. Please also check whether input program id is valid.

Exec search\_course(2019, ‘second’, ‘CS 325’);

Feature 5: compute number of classes assigned to the instructor

Input: courses, instructor, year, semester

Output: check the number of scheduled courses taught by the instructor with course load if it is applicable

Exec assigned\_courses(‘IS 620’, ‘Chen’, 2019, ‘second’);

Feature 6: Assign the courses to the instructor who is willing to take

Input: id of courses, year, and semester, the instructor who willing to take course

Output: check if course needs more sections if so assign new sections to instructor who is willing to teach and create schedule for this course. Otherwise print a message with enough sections.

Exec course\_instructor(‘DS 421’, 2019, ‘third’, ‘Chen’);

Feature 7: Assign courses in a department

Input: instructor name, course id, year, semester, course load

Output: it will check whether all instructor has been assigned enough courses and print the names of instructors who have not

Exec course\_department(‘Chen’, ‘CS 520’, 2019, ‘second’, 1);

Feature 8: Assign rooms and time to a scheduled section

Input: schedule id

Output: Check whether schedule id is valid or not. Then check the scheduled section already has a room and time block, if so print an message or find a room and a time block pair that specify the conditions.

Exec assign\_room(‘SCH001’);

Feature 9: Assign rooms and time blocks for courses

Input: department id, year, and semester

Output: will go through every scheduled section in that department, year, and semester and assign for each section a room and time block

Exec assign\_timeblocks(‘IS’, 2019, ‘second’);

**Registration Features**

Feature 10: special permission for a student and a scheduled section

Input: student id and schedule id

Output: if both are valid then check for the special permission that includes register the class without any prerequisite subject or can register in the full class too

Exec permission(5, ‘SCH004’);

Feature 11: prerequisite of a class

Input: schedule id and student id

Output: check that student needs any prerequisite class or not or the student has already taken the class

Exec prerequisite(‘SCH004’, 5);

Feature 12: allow student to register for a course

Input: schedule id, student id

Output: check the certain condition if the schedule id and student id are valid and also check is student going to take a class another time if he got D grade also check if he need any prerequisite class and can register with the special permission also check class is still open or not and can enter with the permission

Exec register\_course(‘SCH004’, 5);

Feature 13: allow the student to drop the course

Input: student id and schedule id

Output: student can drop the class and it this happen then the student who is in the waiting list, will automatically enroll in the class and if there is no one in the waiting list and if class was closed then change the status of the class from close to open

Exec drop\_course(5, ‘SCH004’);

Feature 14: allow the student to print the course schedule

Input: student id, year, semester

Output: course id, course name, section id, status

Exec print\_schedule(5, 2019,’second’);

**Statistic Features**

Feature 15: print enrollment statistic for a department

Input: department id, year, semester

Output: print the total numbers of students registered in the particular class also print the total courses in the department and also print the student name who are in the waiting list along with the course id, course name, section id

Exec statistic(‘CS’, 2019, ‘second’);

Feature 16: identify the top courses with the longest waiting list, the fewest scheduled class section, and time blocks that have fewest classes

Input: year, semester, k

Output: print the id, name of classes, length of waiting list, room id, room name, scheduled class section, block id, days of the week, start time

Exec longest(2019, ‘second’, 4);