## Time Instructor **UML Class Diagram** - year: int - instructorID: str - month: int - courses: courseID∏ Server - date: int - name: str - hour: int - Course - minute: int - Student + getName(): str - Instructor + setName(name: str): boolean + setTime(int, int, int, int, int): boolean + setInstructorID(instructorID: str): boolean + createStudent(studentID: str, name: str): boolean + getInstructorID(): str + createInstructor(instructorID: str, name: str): boolean + setCourses(courses: str[]): boolean + findStudent(studentID: str): Student + getCourses(): courseID[] + getCourses(studentID: str): courseID[]: str[] + addCourse(courseID: str): boolean + addCourse(studentID: str, courseID: str): boolean + deleteCourse(courseID: str): boolean + removeCourse(studentID: str, courseID: str): boolean + deleteTask(instructorID: str. courseID: str. + findInstructor(instructorID: str): Instructor taskID: str): boolean + findCourse(courseID: str): Course + updateTask(instructorID: str, courseID: str, + createCourse(instructorID: str, courseID: str, courseName: str): type: str, deadline: Time, taskID: str, blurb: boolean str): boolean + getCourseInst(instructorID: str): courseID[]: str[] + createTask(instructorID: str, courseID: str, + createTask(instructorID: str, courseID: str, type: str, deadline: type: str, deadline: Time, taskID: str, blurb: Time, taskID: str, blurb: str): boolean str): boolean + updateTask(instructorID: str, courseID: str, type: str, deadline: Time, taskID: str, blurb: str): boolean + deleteTask(instructorID: str, courseID: str, taskID: str): boolean Task + allCourses(): courseID[]: str[] - type: str - deadline: Time - taskID: str - blurb: str + setType(type: str):boolean + getType(): str + setDeadline(year: int, month: int, date: int, Course hour: int, minute: int): boolean + getDeadline(): Time - tasks: Task[] + setTaskID(taskID: str): boolean - taskoibs: + getTaskID(): str - admins: str∏ + setBlurb(blurb: str): boolean - courseID: str Student + getBlurb(): str - name: str - studentID: str + setName(name: str): void - name: str + getName(): str courses: courseID[] + setCourseID(courseID: str): void + getCourseID(): str + deleteTask(taskID: str): void + setStudentID(studentID: str): void + setTasks(tasks: Task[]): void + getStudentID(): str + getTasks(): Task[] + getCourses(): courseID[] + addTask(Task): void + addCourse(): void + deleteTask(taskID: str): void + deleteCourse(courseID: str): boolean + findTask(taskID): Task + setAdmins(admins: str[]): void + getAdmins(): str[] + addAdmin(adminID: str): void + deleteAdmins(adminID: str): void