

<b>Student {abstract}</b>
#name: String #age: int #gender: String #year: int #major: String #gpa: double #roommatePreferences: List<String> #previousInternships: List<String>
+calculateConnectionStrength(other: Student): int {abstract}



<b>UniversityStudent</b>
+UniversityStudent(name: String, age: int, gender: String, year: int, major: String, gpa: double, roommatePreferences: List<String>, previousInternships: List<String>) +calculateConnectionStrength(other: Student): int

<b>StudentGraph</b>
-adjacencyList: Map<UniversityStudent, List<Edge>>
+StudentGraph(students: List<UniversityStudent>) +addEdge(one: UniversityStudent, two: UniversityStudent, weight: int): void +getNeighbors(student: UniversityStudent): List<Edge> +getAllNodes(): Set<UniversityStudent>



<b>Edge</b>
-otherStudent: UniversityStudent -weight: int
+Edge(otherStudent: UniversityStudent, weight: int) +getOtherStudent(): UniversityStudent +getWeight(): int

<b>DataParser</b>
+parseStudents(filename: String): List<UniversityStudent>

<b>GaleShapley</b>
+assignRoommates(students: List<UniversityStudent>): void

ReferralPathFinder
+ReferralPathFinder(graph: StudentGraph) +findReferralPath(start: UniversityStudent, targetCompany: String): List<UniversityStudent>

PodFormation
+PodFormation(graph: studentGraph) +formPods(podSize: int): void

Main
+main(args: String[]): void

