1. Using the Default Pattern for custom type definition discussed in class, implement a "class" called Student. Each student has a name, number of credits, and an array of integers representing the courses the student takes. The Student type has a common field named "college" initialized to "UMCP". The type has a method called "info" that displays the student's information (see example below for the format to use). The following example relies on the type you are expected to define:

```javascript
var s1 = new Student("Kelly", 15, [414, 420]);
s1.info();
```

Output:
```
Name: Kelly, Credits: 15, Courses: 414,420, College: UMCP
```

Answer/Grading(42 pts):

```javascript
function Student(name, credits, courses) {
  // (4 pts) for prototype
  this.name = name;  // (3 pts)
  this.credits = credits;  // (3 pts)
  this.courses = courses;  // (3 pts)
}

Student.prototype = {
  // (3 pts)
  constructor: Student,
  // (3 pts)
  college: "UMCP",
  // (3 pts)
  info: function() {
    // (3 pts)
    document.writeln("Name: ", this.name);  // (4 pts)
    document.writeln("Credits: ", this.credits);  // (4 pts)
    document.writeln("Courses: ", this.courses + " ");  // (4 pts)
    document.writeln("College: ", this.college + "<br/>");  // (5 pts)
  }
};
```
2. Define a “derived” class called GradStudent based on the Student type above. A graduate student has all the values associated with a student, but in addition it has an advisor (string). A method getAdvisor that returns the advisor is associated with this custom type. The following example relies on this type:

```javascript
var gs1 = new GradStudent("Mike", 15, [414, 420], "Dr. Smith");
gs1.info();
document.writeln(gs1.getAdvisor());
```

Output:
Name: Mike, Credits: 15, Courses: 414,420, College: UMCP
Dr. Smith

Answer/Grading (30 pts):

```javascript
function GradStudent(name, credits, courses, advisor) { // (6 pts)
    Student.call(this, name, credits, courses); // (7 pts)
    this.advisor = advisor; // (3 pts)
}
GradStudent.prototype = new Student(); // (6 pts)
GradStudent.prototype.getAdvisor = function() { return this.advisor; } // (8 pts)
```