```
function  [courses] = make_course_struct(course_cell)

% Map letter grade to credit.  I assumed a
% class in progress is an A. We will handle
% the case of non-letter grade classes later.
grade_map = {'A+',4.3;'A',4.0;'A-',3.7;...
             'B+',3.3;'B',3.0;'B-',2.7;...
             'C+',2.3;'C',2.0;'C-',1.7;...
             'D+',1.3;'D',1.0;'D-',0.7;...
             'NP',0.0;'P',1.0;'CR',1.0;...
             'NC',0.0;'S',1.0;' ',4.0};

nclass = length(course_cell);

% Convert to convenient structure
courses = struct('department',cell(nclass,1),...
    'course_num',cell(nclass,1),...
    'type',cell(nclass,1),...
    'grade',cell(nclass,1),...
    'grad_units',cell(nclass,1),...
    'gpa_units',cell(nclass,1),...
    'gpa_credits',cell(nclass,1),...
    'quarter',cell(nclass,1),...
    'year',cell(nclass,1),...
    'academic_year',cell(nclass,1));

% Make grades structure
first_year = inf;
for i = 1:nclass
    % Department (extract using regular expression and known format of
 course)
    % There are a number of ways to do this.  Regular expressions not
 required.
    courses(i).department = course_cell{i,1}
(1:regexp(course_cell{i,1},'\s')-1);

    % Course number (extract using reglar expression and known format)
    course_num_ind = regexp(course_cell{i,1},'\d');
    courses(i).course_num = str2double(course_cell{i,1}
(course_num_ind));

    % Course letter (extract using reglar expression and known
 format).
    % If course number is followed by letter, i.e. ME335A, ME335B,
 ME335C.
    courses(i).course_letter = [];
    if course_num_ind(end) < length(course_cell{i,1})
        courses(i).course_letter = course_cell{i,1}
(course_num_ind(end)+1:end);
    end

    % Grade
    courses(i).grade = course_cell{i,6};
    if isempty(courses(i).grade)
        courses(i).grade = 'IP';
    end

    % Grade type (letter or P/NP or S/NP, etc)
    courses(i).type = course_cell{i,5}(1);

    % Units
    courses(i).units = course_cell{i,4};

    % Graduation/GPA credits
    % Only LETTER grades get counted towards GPA
    % Only count classes that are COMPLETED
    courses(i).grad_units = 0;
    courses(i).gpa_credits= 0;
    courses(i).gpa_units  = 0;
    if ~strcmpi(courses(i).grade,'IP')
        courses(i).grad_units = courses(i).units;
        if courses(i).type == 'L'
            courses(i).gpa_credits =
 map_grades_to_credits(course_cell(i,:),grade_map);
            courses(i).gpa_units = courses(i).units;
        end
    end

    % When taken (extract year and quarter that class was taken)
    % I used regular expressions. Could have just used string
 expressions.
    quarter = course_cell{i,3}(regexp(course_cell{i,3},'\s')+1:end);
    year_ind = regexp(course_cell{i,3},'\d');
    if strcmpi(quarter,'Autumn')
        year = str2double(course_cell{i,3}(year_ind(1:4)));
        % First academic year defined as the autumn your first class
        % at Stanford was taken.
        first_year = min(year,first_year);
    else
        year = str2double(course_cell{i,3}(year_ind(5:8)));
    end
    courses(i).quarter = quarter;
    courses(i).year    = year;
end

% Determine academic year (i.e. number of years at Stanford)
for i = 1:length(courses)
    courses(i).academic_year = courses(i).year - first_year;
    if strcmpi(courses(i).quarter,'Autumn')
        courses(i).academic_year = courses(i).academic_year + 1;
    end
end


end
```

```
function  [grade_credit] = map_grades_to_credits(grade,grade_map)
% Map grade to credits obtained for the grade

grade_credit = zeros(size(grade,1),1);
for i = 1:size(grade,1)
    %units = grade{i,4};
    grade_earned  = grade{i,6};
    grade_map_ind = ismember(grade_map(:,1),grade_earned);
    grade_credit(i) = grade_map{grade_map_ind,2};%*units;
end

end
```

```
Not enough input arguments.

Error in make_course_struct (line 13)
nclass = length(course_cell);
```

Published with MATLAB® R2019a