

## COP 4516: Problem Solving Techniques and Team Dynamics Syllabus

**UPDATED AS OF MARCH 1, 2018!!!**

Course Website: <http://www.cs.ucf.edu/~dmarino/progcontests/cop4516/spr2018>

Lecturer: Arup Guha

Email: [dmarino@cs.ucf.edu](mailto:dmarino@cs.ucf.edu)

Class Times/Locations: T 10:30 – 11:20am (HEC-111),

F 9:00 – 11:50am (HEC-202, HEC-308)

Office: HEC – 240

Office Phone Number: 407-823-1062

Office Hours: TBA (check course website)

**I do NOT check my WebCourses email. Please email me at [dmarino@cs.ucf.edu](mailto:dmarino@cs.ucf.edu) to contact me.**

**This course requires Friday attendance. Please don't plan trips where you leave on Thursday night.**

**Course Description:** This course covers training similar to that given to UCF's programming team. Lectures will cover classical algorithms, most of which are taught in Computer Science 2, that tend to be useful in solving programming contest problems. Emphasis will be placed on implementation issues. The general topics covered are: number theory, brute force search, greedy algorithms, graph algorithms, dynamic programming algorithms and geometry algorithms.

**Note: There is NO course textbook. Rather, course notes and websites will be used as primary sources. If one strongly desires a book, here are a few that would suffice:**

Introduction to Algorithms – Cormen, Leiserson, Rivest, Stein (ISBN: 978-0-262-03384-8)

Programming Challenges – Skiena, Revilla (ISBN: 0-387-00163-8)

Algorithms – Dasgupta, Papadimitriou, Vazirani (ISBN: 0-07-352349-2)

The Design and Analysis of Algorithms – Levitin (ISBN: 0-321-35828-7)

Competitive Programming 3 - Halim and Halim (cpbook.net)

## Grading

This course will have five components

Item	Quantity	Total Percentage
Hackpack	1 (in teams)	10
Online Contest Participation	2	10
Individual Contests	6	30
Team Contests	6	30
Individual Contest Exam	1	10
Team Contest Exam	1	10

### Hackpack

Each team will be responsible for producing one hackpack. Minimal requirements will be given for the hackpack. If these are met, 80% will be awarded. The last 20% will be unspecified and based on exceeding the minimum requirements. Each team will make a single submission of their hackpack towards the end of the course, but it's strongly suggested that each team create weekly internal deadlines for the hackpack.

### Online Contest Participation

There are several online websites that run programming contests. Some of these are: CodeForces, USACO and CodeChef. You must show me the scoreboard on your laptop during my office hours for one contest **before** March 2, 2018 and for a second contest **before** April 20, 2018 to get credit for this portion of the course grade. If you want to compete on a site other than these, please tell me the site and I'll let you know whether or not I approve it. In order to get full credit for this component of the course, I will require that you get at least one problem correct in contest. (**Note: for those of you with past contest experience, I may place the bar a bit higher for you to get full credit based on what I know about your performances in past contests. For students who intentionally underperformed in the past, I gave 60% or 80% for this grade.**) Also, UCF runs weekly Saturday practices. I'll allow credit for this portion of the class for attending a UCF practice. ***To gain the credit for a UCF practice, you must stay for 3 of the 5 hours.***

### Individual Contests

For the first six weeks of the course, individual contests will be held on Fridays, each with three or four problems. 100% will be given to any question correctly submitted during the contest. 80% will be given to any question correctly submitted after the contest, before the test data is posted. 60% will be given to any question correctly submitted after the test data is posted. No credit will be given for questions not correctly solved by the Wednesday morning (10 am) after the contest has completed.

### Team Contests

For the second half of the course, each Friday competition will be in teams. Grading will be determined in the same manner as the individual contests and the same grade will be assigned to each team member, regardless of who solves which questions. **Note: It's rarely the case on a good team that team members split up the work equally. Trying too hard to split up the work equally will likely worsen your team's performance and grade.**

### Individual Contest Exam

After the first six weeks of the course, a more comprehensive contest will be given for individuals. The only differences are that no submissions will be allowed after the contest is over, that this contest counts towards 10% of the final grade instead of 5% and all of the questions in this contest will be newly created for the contest itself. **Partial credit will be given for the last incorrect submission for any problem not properly solved in contest since there is no late deadline for submission. This means you'll have to alter your strategy for this one contest to make sure you have base code written for all problems.** As usual, any problems completed during the contest will automatically receive 100%.

### Team Contest Exam

During the final exam period for the course, the teams will compete in a final contest, which will be designed to be easily solvable with a good hackpack. No submissions will be allowed after the contest has ended **but partial credit will be given to the last incorrect submission, similar to the Individual Contest Exam.** As usual, any problems completed during the contest will automatically receive 100%.

### Method of Awarding Final Grades

Unlike other courses, final grades aren't awarded solely on the basis of the percentage in the course. Since this is a contest class and I have to award grades to individuals, even though a bulk of the grade comes from teamwork, I don't want to award a grade to someone that was largely earned due to the excellence of a teammate. To that end, for each letter grade cut-off, I'll set a minimum number of problems solved **in the seven individual contests** in addition to the usual percentage cut-off. In order to earn a letter grade for the course, a student must meet **BOTH** cut-offs. Unfortunately, I won't announce these cut-offs until the end of the semester as I feel that doing so would promote some students to produce substandard work. In the past, I've changed my cut-offs for individual contests (made them lower) because I've seen some individuals work extremely hard in the team phase of the class. I would hate to tell a student that the best grade they could earn is a C and then see them slack in the second part of the course. To give you an idea where this cut-off might lie, in the past, it's usually been around 12-15 problems for an A, over the seven individual contests, so an average of close to 2 problems correct per contest should be good enough to earn an A in the course. **Note: Plus/Minus grades will be awarded when deemed appropriate.**

### **Policy for Absences from Friday Contests**

For individual contests, if you are absent for a significant portion (more than 1 hour) of the contest **without prior approval**, then **NO PARTIAL CREDIT CAN BE EARNED FOR SOLVING PROBLEMS AFTER THE CONTEST**. (Naturally, if you solve all of the problems in the contest early, you are immediately free to go! But, if you show up very late or leave very early and leave problems unsolved, then you can't later earn 80%/60% credit on those questions.)

Since many of you are busy (whether it be lining up interviews for potential jobs or working current jobs), if you know in advance that you'll have to miss on a particular Friday for an individual contest, then **please notify me in advance**. I will let you make up the grade by running a specified online contest in real time. (**I will give you a short selection of options and you must choose one of them.**) This is in addition to the two online contests you'll have to run during the semester. Based on your performance and the code you show me, I'll assign a grade that I feel is appropriate. I need to leave myself full discretion here due to the differences in difficulty of various contests and the limited availability of online contests during short time frames.

If you need to miss a team contest Friday **and tell me in advance**, you have two options: (a) If your team agrees, they can pick up your slack and I'll give you the grade they earn for the week without you. (b) I can assign you an additional individual online contest to run.

### **Academic Misconduct Policy**

Since this is an elective (you don't have to be here, so I assume you are here because you WANT to be), I will be more harsh with academic misconduct than usual. **In particular, if there are any clear violations of the academic misconduct policy, I will make official documentation with the necessary witnesses, record the transgression with UCF and fail you from the course.** The rules for the course are as follows:

- 1) During any individual contests, individuals may ONLY look at language APIs online and no other electronic materials. Students may look at any printed materials. Students **may not** talk to any other students during the individual contests about any items that I might think may be helpful in solving the problems. I reserve my right to use my discretion on whether or not a topic of conversation may be helpful in solving a problem. You are safe in telling someone where the bathroom is or describing where a restaurant is located, for example. You are NOT safe in explaining the steps of any algorithm or pointing out a restriction in a problem, for example.
- 2) During team contests, you may only talk to your team members about problem related issues and you may ONLY look at language APIs online and no other electronic materials. You may look at any printed materials. Communication with non-team members in the course is limited as previously described.
- 3) Backpack – you may look at algorithmic descriptions in print or electronically in developing your team's backpack, but **ALL** of the code must be written originally by a combination of your team members. It is perfectly fine if some or all of the code is written individually, or if all the code is produced with each team member present. It's likely that some code will be written individually and that teams will meet to “finalize” algorithms in their backpack periodically so that each team member is comfortable using any part of the backpack.

## Tentative Schedule

Week	Tuesday Class	Friday Class
Jan 8	Java API, Brute Force	Ind Contest #1
Jan 15	Java API, Greedy	Ind Contest #2
Jan 22	Java API, Trees	Ind Contest #3
Jan 29	Graph – DFS, BFS	Ind Contest #4
Feb 5	Shortest Distance, Top Sort	Ind Contest #5
Feb 12	Cumulative Sums, Mathematics	Ind Contest #6
Feb 19	Exam Review	Ind Contest Exam
Feb 26	Team Dynamics	Team Contest #1
Mar 5	Graph – Network Flow	Team Contest #2
Mar 12	<b>Spring</b>	<b>Break!!!</b>
Mar 19	DP – Take it or leave it (Knapsack, LCS, Neighbor) (Guest Lecture)	Team Contest #3
Mar 26	DP – Edit Distance, MCM, World Series, Traveling Salesman DP	Team Contest #4
Apr 2	Geometry	Team Contest #5
Apr 9	Binary Search Applications	Team Contest #6
Apr 16	Guest Lecture - Binary Index Trees	<b>FRIDAY CLASS CANCELED</b>
May 1	<b>Team Contest Exam (10 am – 1 pm)</b>	

I may change this schedule, thus class attendance is important. This is a general time frame only and is subject to the needs of the class. At the end of each class I will tell you what we will be discussing during the next class period. I may not post formal notes from the lectures, so please take all necessary notes during lectures. Good notes for most of the topics I will cover can be found online. I will make whatever notes to which I have access available online for students.