SCUBA dive planning assignment
As of 16 February 2015

DISCLAIMER: on every interface you develop you MUST have a statement saying your system is a PROTOTYPE and cannot be used to plan real dives. Also, in the beginning comments of every function/method/object you MUST have a statement saying the systems is a PROTYPE and cannot be used to plan real dives. This is serious. You need to make it completely clear to anyone who sees your code, your interfaces, or interacts with your prototype that your system is not to be trusted with their lives.

Check-in 1 – to be done in teams
Due: end of lab Friday 20 February 2015.

Recreational SCUBA diving is a popular sport that incorporates interesting physics. In the past, divers used paper-and-pencil and "dive tables" to plan safe dives. What would make a dive unsafe? Coming up too fast can give you "the bends", a potentially debilitating, sometimes fatal, phenomenon involving the gasses we breathe everyday interacting with our blood at different pressures.

Clearly, divers need to plan their dives to avoid problems. Fortunately, there are modern dive computers that assist with pre-dive planning and real-time monitoring during dives. You will have an opportunity to develop dive planning software.

Your first task is background research, then answer questions. You can skim the background readings since some are long and dense, but I want you to get a feel for what is involved with dive planning. Feel free to talk to divers and search the web for resources.

Background reading:
Wikipedia page about dive profiles: http://en.wikipedia.org/wiki/Dive_profile
Dive Tables Explained: http://www.scubadiverinfo.com/2_divetables.html
Recreational Dive Planner:
http://elearning.padi.com/company0/tools/RDP%20InsforUseMet.pdf

Deliverable:
One document per team, each team member turns in a copy of the team document.

One page pdf document with explanations in your own words of the following terms (one paragraph for each term is enough):

1. Dive profile
2. Decompression sickness
3. Safety (or decompression) stop
4. Bottom time
5. Surface interval
6. Nitrogen narcosis

Check-in 2 – to be done in teams
Delverable due: Friday 20 March 2015

Subtask for Lab Wednesday 25 February
Each team member, using no more than two pages, sketch the main interface(s) for a prototype dive planner. This is not the same as a dive computer that you wear during the dive itself. Your sketches should be in a form usable for paper prototyping, although you will not be experimenting with users– just create the sketches. Feel free to talk with your teammates, but generate the two pages individually.

Subtask for Lab Friday 27 February
As a team, come up with one paper prototype for your team. You will not be running test users (unless you want to). Design both a web interface and an app.

Subtask for Labs 4, 6 March
As a team, decide on your software design, description of the software process you will use, language and where you are hosting your project (github, google code, …), estimated LOC, and your approach to V&V (verification and validation).

Subtask for Lab on 11, 13 March
You will have a F2F check in with the TA where you will demo *some* implemented functionality (not the interface). This can be as simple as implementing a dive table. You will also show him where you are hosting your code (and other deliverables if you choose to), development environment (probably Eclipse), and the version control system you chose (git, subversion).

Deliverables due **Friday 20 March**
Each member of a team turns in the same deliverable: No more than **five page pdf** document including:

- Samples of interface for web, and samples of interface for app.
- Software design (your choice of notation, it can be as simple as "boxes and arrows") of the major functionality.
- Description of your software process. I recommend using TDD and short sprints. You might also want to do code inspections as a team.
- Estimated LOC for implementing functionality (not interface).
- Estimate of LOC for interface.
- At least one paragraph describing your V&V (verification and validation) strategy. Since this is safety critical software you need a convincing test strategy.

Check-in 3 – to be done in teams
Deliverable due: **Friday 17 April**

Subtask for Lab 25, 27 March
You will have a F2F check in with the TA where you will demonstrate your interface (the functionality can be incomplete).

Subtask for Lab 8, 10 April
You will have a F2F check in with the TA where you will discuss your V&V plan and activities.

Deliverable due Friday 17 April
Each member of a team turns in the same deliverables: No more than **two page pdf** document including:

- Revised LOC estimate for functionality.
  Revised LOC estimate for interface.
- At least one paragraph describing V&V activities to-date, particularly number of test cases run and number of defects found.
- Brief description of any deviations from your planned design or process reported the previous check-ins.

Demo and final deliverables – to be done in teams
Deliverable due: **Friday 1 May**

Each member of a team turns in the same deliverable: No more than **two page pdf** document including:

- URL of a narrated video (five minute maximum) demonstrating the functionality of your system. The video must be on YouTube or in mpeg 4 format in a publicly accessible place (like a Public folder on dropbox, box, or google docs). You can use a free screen capture system like jing, screencast-o-matic, or QuickTime if you are running OS X.
- URL of code repository where I can check the commits and see the source code.
- Brief summary of V&V activities. You need to convince me that your testing was adequate (or not) for a safety critical system.
- Project post-mortem, i.e., if you were starting over, what would your team do differently.