1 Executive Summary

In Technical Memo 2 your team will break your project down into components, analyze the criteria that characterize an ideal solution to each component, and assesses the relative importance of potentially conflicting objectives. In drafting this memo, your team should arrive at a set of concrete goals for the project.

Each team should post a .pdf copy of their technical memo on Canvas by 11:59pm on Thursday 21 September 2017.

2 Content of the Memo

Technical Memo 2 should contain (1) an executive summary; (2) a description of the main components of your project (with a diagram); (3) an identification and quantitative description of the project criteria; (4) a prioritization of the different objectives based on pairwise comparisons.

Your client has a specific objective for the project at hand, which you identified in Technical Memo 1. The design of a solution that meets this objective will be challenging; if you plunge directly into the problem now, you may well focus on secondary priorities, while neglecting or postponing the most critical aspects required of a successful project. Careful planning is necessary to avoid such missteps. At this stage, we thus carefully enumerate, quantify, and evaluate the criteria for a successful project.

2.1 Project Components and Ideal Criteria

Each project raises different goals and challenges. You should very specifically tailor this memo to the considerations of your particular project, as informed by discussions with your client. To illustrate the process we will walk through a generic example here; you should develop your own criteria that fit your own problem. There might be some overlap, but you should not merely copy the example given here.

Step 1: Divide your problem into components

Your project is likely to has several distinct components; start by subdividing your problem into these components, assess the criteria required for each part, then weigh the relative importance of each part.

- Identify several concrete sub-problems that you would ideally solve, and prioritize them: Which is most important? How do the components depend on one another? Are there common elements that will enable you to address several sub-problems at once?
Consider a fictitious project that involves modeling earthquakes, for which we identify four main components:

(1) Mathematical modeling;
(2) Data collection;
(3) Data analytics;
(4) Visualization.

We assess that (1) and (2) can be completed independently, but both are necessary to complete step (3). Step (4) can be started after part (2) is complete, but a full solution will also require the output from step (3).

Please illustrate these dependencies in a simple diagram.

```
(1) Math modeling       (3) Data analytics       (4) Visualization
       \           \         /                \\
(2) Data collection     /         /                \\
       /         /                \\
```

This crude figure was drawn in \textit{\LaTeX}’s “picture mode”. Many fancier and more flexible options are possible, such as using the \texttt{pgf} (portable graphics format) \LaTeX package, or importing an image from a drawing program.

**Step 2: Establish criteria for each component, and quantify them**

For each component of your project, establish a list of criteria that characterize an ideal solution. Be quantitative wherever possible. (We will be looking for such quantifications when we evaluate your memo.)

Here are a few examples using components from the fictitious project above.

Example criteria for (1) Mathematical Modeling

- \textit{Language}. Code should be in a language that our client specifies (MATLAB or Python).
- \textit{Physical fidelity}. The model should incorporate three-dimensional physics.
- \textit{Speed}. The simulation should run in less than 4 hours on a commodity laptop.
- \textit{Accuracy}. Numerical approximations should provide answers that have at most 0.1% error from the exact answers.
Notice that we have not specified any specific algorithms here. Those will be determined at a later stage. Indeed, these design criteria are meant to guide our eventual choice of algorithm.

Also notice that some of the design criteria conflict. Physical fidelity and accuracy both work against speed. Conflicting criteria are fine at this stage: the scoring procedure we describe below will help you discern priorities in the face of such conflicts.

Example criteria for (2) Data collection.

- **Volume of data.** We seek to locate data for at least ten different earthquakes of magnitude 7.0 or higher, measuring vibrations at the observation station every milisecond.

- **Veracity of data.** We seek data from instruments that record vibrations with at least 0.1% accuracy.

- **Number of data sources.** To ensure data independence, we seek data from at least five different measuring stations.

- **Data format.** We seek data in `.csv` format.

- **Access.** We seek only freely available public data sources.

Notice that we have not listed specific data sources here. However, to give realistic quantitative goals, you would likely need to do some preliminary research to get an idea of what resources are available.

Each team should spend some time brainstorming the criteria for the project. These might not be immediately obvious, so think broadly and deeply. Please do not just copy the criteria in the examples above, though you might well incorporate some of them into your list.

Teams are expected to identify roughly four to six criteria for each component of the project, and quantify them as best as possible.

**Some tips on quantification.**

Students typically begin with criteria that are entirely vague: the simulation must be “fast” or “accurate”. These terms have different meanings for different people, and in different settings. If left alone, they are virtually meaningless; you will not be able to judge whether your eventual solution meets such vague criteria. We endow meaning by associating each criterion with some quantification, which gives a goal to strive for and a metric for later evaluating success.

You should thus replace “fast” by, e.g., “One complete simulation runs in less than 10 seconds in MATLAB on my MacBookPro laptop”.

You should thus replace “accurate” by, e.g., “The computed solution is within 10% of the optimal value” or “The algorithm should correctly identify tuberculosis bacteria in at least 95% of cases, and give false positives in no more than 10% of cases.”

Some design criteria are difficult to quantify. In this case, you might need to settle for a relative comparison or a user-defined scale, e.g., “On an ease-of-use scale of 1 (easy) to 5 (hard), a panel of typical users scores our user interface either 1 or 2.”
Coming up with quantitative measures for your design criteria will likely be rather challenging for some projects in this course. Do your best to make sense of this. The more clearly you see your goals, the more likely you are to achieve them.

**Step 3: Weighing your criteria**

The criteria you have identified will likely conflict. For example, it might be difficult to design a solution that is both fast and accurate. To prioritize your efforts in the rest of the project, you should compare the relative merits of the criteria. Engineers do this by a pairwise comparison chart. Build a table with the major criteria listed down each column and across the top.

For example, the component “(1) Mathematical Modeling” above leads to the following table.

<table>
<thead>
<tr>
<th></th>
<th>Physical Fidelity</th>
<th>Speed</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Language</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Physical Fidelity</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Speed</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Accuracy</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

To fill in row \( j \) and column \( k \) of this table, ask yourself: Which is more important: criterion \( j \) or criterion \( k \)? If \( j \) is more important than \( k \), put a 1 in the \((j, k)\) cell; otherwise, put a 0 in that cell.

For example, you might fill in the above chart as follows.

<table>
<thead>
<tr>
<th></th>
<th>Physical Fidelity</th>
<th>Speed</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Language</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Physical Fidelity</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Speed</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Accuracy</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

In this example, we have decided that “Physical Fidelity” is more important than “Accuracy”, meaning that we want a three-dimensional model even if we cannot afford to use a sufficiently fine mesh to give the accuracy required in our quantification of that metric. We have ranked all the other criteria as more important than “Speed”: that means we are willing to sacrifice runtime for the other considerations.

Notice that two teams could have the same design criteria, but rank them differently because the priorities of their different clients.

Now, to prioritize, add a “score” column, and add across the rows.

<table>
<thead>
<tr>
<th></th>
<th>Physical Fidelity</th>
<th>Speed</th>
<th>Accuracy</th>
<th>score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Language</td>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Physical Fidelity</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Speed</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Accuracy</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>
These scores then give an ordered list of our priorities for this component, listing the criteria from the highest score to the lowest.

1. Physical Fidelity
2. Accuracy
3. Language
4. Speed

This result will affect the way you proceed with the rest of your project. In this case, you would direct your efforts to design a model with high Physical Fidelity (i.e., incorporating 3d physics) that is quite accurate; this might mean a compromise in your choice of programming language, or in runtime.

- Repeat this exercise for each of the major components you identified in Step 1. For each component, include your scoring table in your memo, along with the ordered list of the criteria (from most to least important). Important: Provide some basic explanation for the decisions you make as you fill in each scoring table.

By identifying and rationally assessing your criteria, you have shaped the course of your project in a manner that can be logically defended. This decision process helps manage client expectations and sets realistic goals for your team. In the next tech memo, you will brainstorm solutions that address the components you identified here.

3 Grading Rubric

This memo will be graded on a 100 point scale, according to the following rubric.

**identification of components:** 15 points
Identify the key components of the project. Include a diagram that shows their dependencies.

**criteria:** 30 points
Identify 4 to 6 criteria for each major project component. Criteria should be relevant and important. Significant criteria should not be missing.

**quantification of design criteria:** 15 points
Provide a quantitative measure for each criterion.

**evaluation of design criteria:** 20 points
Perform pairwise comparison of design criteria. Present a prioritized list of the criteria. Justify the decisions that led to these scores.

**formatting, style, grammar:** 20 points