Project description Q1 2022

Due Sunday 30-10-2022 at 23:59 on Canvas:
Hand in a pdf-document containing your report.

There are three options for a project. The first option is to participate in a (local) contest by engineering an efficient algorithm for a given well-defined problem. The second option is to tackle a ill-defined problem from scratch. The third option is a free-choice option – you can propose a topic with a group but make sure to get approval before you start.

Below, you'll find details for two options as well as some notes on how your work will be graded. Be sure to also read the section at the end, on requirements on the report, group set-up and on grading.

Note: *If you are retaking this course, you cannot do the same topic again. If you are unsure, please ask.*

Option 1: Contest on embedding boundaries on a grid

Problem setting. In Lecture "Cartograms and grid maps", you will learn about grid maps to schematically portray geographic regions using squares. In this project, we consider a simplified subproblem: how to place the boundary regions, such the regions maintain the right relations to their neighbors.

Formally, let B denote a boundary of n regions r_i, defined by a sequence of $n-1$ directions d_i: $\leftarrow, \rightarrow, \uparrow, \downarrow$. We want to embed this boundary into a unit grid of width $W \geq 1$ and height $H \geq 1$. That is, we aim to assign an integer coordinate (x_i, y_i) to each region r_i. The following constraints define validity of such an embedding:

- Coordinates lie in the given grid space: $0 \leq x_i \leq W$ and $0 \leq y_i \leq H$ for all $i \in [1, n]$.
- Coordinates are unique: $x_i \neq x_j$ or $y_i \neq y_j$ for all $i, j \in [1, n]$ with $i \neq j$.
- Coordinates of subsequent regions differ by exactly one in x-coordinate, or exactly one in y-coordinate (but not both): $|x_i - x_{i+1}| + |y_i - y_{i+1}| = 1$, for all $i \in [1, n)$.

For valid embeddings, we define their quality as the number of deviations from the given directions. That is, if $d_i = \rightarrow$, then $x_{i+1} - x_i = 1$ should hold, otherwise, it counts as a deviation; if $d_i = \downarrow$, then $y_{i+1} - y_i = -1$ should hold, otherwise, it counts as a deviation; etc.

Report. The above gives the hard constraints and the optimization goal for this project. The main idea is to let you tackle a complex algorithmic question. You will need to develop an effective and efficient solution for it, in spite of its computational complexity. There is also a competitive element in the form of a contest (see below). In your report, you may want to pay attention to the following items.

- Literature that your solution builds upon
- Describe your algorithm.
- Analyze properties of your algorithm (e.g. running time, constructed inputs on which it does and does not perform well) and/or of the problem itself (e.g. computational complexity).
- Collect or generate some data, and report on the performance of your algorithm. (*If your algorithm has any parameters that must be set, this is an excellent place to start to phrase a question!*)
- Report on your performance during the contest (and afterwards on the same data if you made improvements!).

Java Framework. On Canvas you will find source code (and a precompiled version) of a Java Framework for the contest. This code provides the basic mechanisms necessary to load problems, save solutions, view solutions, run algorithms and evaluate results, and connect & submit to the contest server. Essentially, you need to implement only an extension to `BoundaryEmbeddingAlgorithm` and add it to the list of algorithms in `AlgorithmList.java` –

a very trivial solution has been added (`ArbitraryAlgorithm.java`) to get you started. More details on how to use the code is provided on Canvas.

Note that this source code is provided in Java only, but you are free to work in the programming language of your own choice. Note, however, that the server interface will work only through the provided Java code – if you do not use the framework, you will need to load the computed solutions into the Java framework via the text format to participate in the contest.

The contest. There are two contests. The first is an ongoing contest, for which you can already get the input. You can automatically upload submissions to the server (see Canvas for details) and score points (see below). This will give you an idea of how you perform and the type of data that the second contest will work with.

One of the lecture slots will be dedicated to the second contest (see Canvas schedule). At the start of the lecture, new data will be made available via Canvas. There tend to be some similar problems (size and structure) as for the ongoing contest. But I often throw a spanner in the works, introducing new challenging datasets – be ready for anything! During the lecture, you can use your algorithm(s) to solve the data sets as well as possible, handing them in via the provided Java framework. At the end of the lecture, the contest server is closed again, so you have only 90 minutes to solve the problem instances.

Input format. The input format is a text file which is formatted as follows:

- the first line contains W and H (in that order), separated by a semicolon.
- the following lines contain characters U(p), D(own), L(eft). R(ight) to indicate a sequence of directions. Note that these can all be on a single line, or split over multiple lines.

Output format. Your solutions are automatically validated and scored in the provided framework or on the server during a contest session. To store your output use the following format:

- each of the n grid points (for $n-1$ directions specified in the input) gives a single line, with their (integer) x- and y-coordinate, separated by a semicolon.

Ranking[1]**.** The results of the contest will be assessed, on an instance-by-instance basis. Each instance awards points as follows, based on the best solution submitted.

- If you have no solutions or all your solutions are invalid, then you score 0 points.
- If your best (valid) solution was the best from all teams, you score 10 points.
- Otherwise, you score $10\sqrt{b}/\sqrt{s}$ where s is your best solution quality and b is the best (valid) quality for this instance over all teams.

[1] To be clear: your rank in the contest does not immediately influence your grade.

Option 2: Sequence maps

In Lecture "Symbol maps and flow maps", we will talk about flow maps. The data this visualizes is essentially a matrix showing flow (migration, transport, etc.) between origin and destination. The detail in this type of movement is very limited. In many scenarios, we can imagine having more refined movement data, where the object being tracked visits a number of locations in a sequence. In this project, your goal is to design a thematic map that adequately visualizes a collection of such sequences.

Report. Below are steps you may take to tackle this problem. You are free to add your own steps and interpretations of this problem, as long as you stay within the sketched topic of visualizing sequences in a spatial context.

1. **Input:** Describe the data you assume as an input: that is, what will be the input of your algorithm?
2. **Validity:** Describe the envisioned solution. Provide and explain one or more constraints that a solution must meet to be valid.
3. **Quality:** Describe one or more intuitive criteria that capture "high-quality", and formalize these by introducing a measure for each, that allows you to measure the quality of a valid solution. Use small examples to explore and illustrate the strengths and weaknesses of your proposed measure(s): judge whether, on that example, the optimal valid solution according to your measure is or is not what you would expect.
4. **Algorithm:** Describe one algorithm (or more) that can generate valid solutions that score well on the measures. Also consider its theoretical properties, e.g.: is this algorithm optimal, an approximation algorithm, or a heuristic? What is its running time?
5. **Experiment:** Generate or collect some data, and perform a brief experiment that answers an interesting question about the problem, your measures and/or your algorithm. A good place to start may be the effect of parameters (of the problem, your measures or algorithms) on the quality of the result: can you find "good default" values that work well in most cases?

Example questions. You are free to follow your own ideas to designing the map and describing the validity and quality of solutions. Here are some suggestions to get you started:

- Do the sequences all start at the same location?
- Can sequences re-visit a location?
- How do you deal with common subsequences in your thematic map?
- What aspects make it easier or harder to see a sequence, in the context of other sequences?

Option 3: Free-choice project

Groups are invited to propose their own topic for modeling and solving algorithmic problems that consider geographic data.

Before you start, make sure to get your topic approved.

Requirements

Groups. The projects are to be done in groups of 5 students. Make sure to register yourself in a group on Canvas, this is necessary to hand in the report.

Deviation from this group size are allowed only after approval of the lecturer.

Report. The main idea for the report is to be in the same form of a conference submission. Thus, your main report should be readable on its own: it contains an introduction to the problem and its relevance, as well as the necessary definitions and notation that you use throughout the report.

Specifically, this means that there is a *strict page limit* of 11 pages including title, author names, and figures. A reference list (if you have any) is not included into this page limit. Any supporting material you want to add that does not fit into the 11 pages, may optionally be added to an appendix (and referred to in the main text); this appendix is placed after the reference list. Nonetheless, your main ideas and findings have to be present in the 11 pages as the appendix is to be read "at the reviewer's discretion". In other words, the report should give a clear view of your work even without the appendix.

Use the LIPICS v2019 template[2] with the `\hideLIPIcs` command. Make sure to provide the name, student number and e-mail address of each group member. You may leave classifiers, keywords and such empty. See the file in Modules on Canvas for a cleaner tex file to start from (start a new project from the template, and override the tex source with the one on Canvas); you may of course add and remove sections as you see fit.

The report has to be typeset in English and submitted as a pdf-document. *Reports not handed in as pdf-document (e.g., as word-documents) score 0 points.*

Present. Each group is required to "present" their ideas. The main idea is to be able to see what projects have been done, and how they approached their problem. Due to the many groups, we cannot unfortunately do actual presentations; instead, you submit (one per group) a small deck of slides that explains your key ideas. These will be shared through Canvas such that you can have a look. See the project introduction slides for an example and the Canvas assignment for details.

The slide deck is not graded, but is a prerequisite: each group needs to submit one slide deck (that's at least semi-informative) in order to receive a grade.

Group grade. The rubric at the end should give you an idea of the assessment criteria. Note that the rubric does not immediately map onto points as it depends on the option you choose. Option 2 relates to all lines, whereas Option 1 does not have a "Modeling" part. Moreover, you may also e.g. emphasize modeling with various considerations, or have simpler modeling with more extensive experiments. For a free-choice project, it depends on the type of project but generally should allow assessment within the given rubric.

Individual grade. A grade for an individual student is, after rounding and maximizing to ten, given by the group grade plus a modifier based on (the last round of) peer review. However, if the modifier is an X, the situation will be examined and the individual grade may deviate from this scheme. If the student's contribution is eventually considered insufficient, the individual grade will be "FL" (failed).

[2] `https://www.overleaf.com/latex/templates/lipics-v2019-sample/gqgybwgdpbpq`

Peer review. Each student judges the contribution of all of his fellow team members with one of the following grades:

+1 considerably higher than average;
0 about average;
-1 considerably less than average;
X inadequate.

The grades per grader are normalized to have average zero (excluding Xs). The modifier of a student is then the average of all normalized grades given to this student. However, if a student scores an X, the modifier is also X.

Peer review is confidential, that is, students will not be told who gave them what grade. Only the results of the last peer review are taken into account. Earlier peer reviews are meant to signal problems. If you have concerns about your or a team members contribution to the project, talk to your team members or contact the lecturer.

Forms for peer review will be provided via Canvas.

		Excellent	Good	Sufficient	Insufficient
General	*Writing*	The writing is easy to read and understand. The structure of the document is logical and flows well. It does not rely on the project description.	The writing is easy to understand. The structure is good. It relies a bit on the project description.	Some effort is needed to understand the writing. The structure could use improvement. It requires the project description to be read first.	The writing is unclear, preventing the understanding of the report's content.
	Critical thinking	The relevant choices are clearly explained. Advantages and disadvantages of alternatives are discussed.	Drawbacks and advantages of the presented solutions (models, algorithms) are clearly discussed.	There is some motivation for choices, without paying attention to the potential consequences of such choices.	No attention is paid the choices made in the project and their effect on the various steps.
Modeling (Option 2/3)	*Validity*	The constraints are very well described, both intuitively and using formalism as appropriate. There are no obvious flaws in the constraints.	The constraints are well described. There are no obvious flaws in the constraints.	The constraints are reasonably well described. There might be some minor flaws in the constraints.	The constraints are not clearly described. There are flaws in the constraints that limit the validity.
	Measures	The measures are very well described, both intuitively and using formalism as appropriate. There are clear examples illustrating and discussing advantages and drawbacks.	The measures are well described. There are examples illustrating and discussing advantages and drawbacks.	The measures are reasonably well described. There is some discussion of advantages or drawbacks.	The measures are not clearly described. There is no discussion on the advantages or drawbacks.
Algorithms	*Algorithm*	The algorithm is very well described and is an excellent solution to the problem.	The algorithm is well described and is a good solution to the problem.	The algorithm is reasonably well described and is a reasonable solution to the problem.	The algorithm is not clearly described or is not a reasonable solution to the problem.
	Analysis	An extensive analysis of the algorithm's (and/or the problem's) properties is included.	A clear analysis of the algorithm's (and/or the problem's) properties is included.	Some analysis of the algorithm's (and/or the problem's) properties is included.	No analysis of the algorithm's (and/or the problem's) properties is included.
Experiments	*Question*	The experiment is very well described and clearly aims to answer a clear and relevant question.	The experiment is well described and aims to answer a clear question.	The experiment is reasonably well described but the question it answers is unclear.	The experiment is unclear or is not answering a question that relates to the problem.
	Data	Realistic and interesting data is used, in addition to the contest data.	Interesting data is used, in addition to the contest data.	Data is not general enough or restricted to contest data.	Data is obstructing relevance of the experiment.
	Conclusion	Correct and clear conclusions are drawn from the experiment to answer the question.	Correct conclusions can be drawn from the experiment to answer the question.	Correct conclusions are drawn, but might not fully answer the original question.	Incorrect conclusions are drawn or the conclusions do not answer the original question.