Coursework commentaries 2016–17

CO2227 Interactive multimedia

Introduction

This year's coursework assignments involved practical implementation of algorithms that form part of the subject material for this course. For both coursework assignments an in-depth investigation of a particular algorithm was required, followed by the development of an artefact based on that investigation. For the first coursework the area of consideration was distance metrics. Students were required to discuss the Levenshtein algorithm, develop an implementation of it for a particular purpose and then develop their own artefact inspired by what they had found in their investigation. This was then rounded off by an academic exercise: to apply the distance metric work in the context of recommender systems. For the second coursework, the subject was inbetweening in the context of animation. This also required a discussion of the area, followed by implementation of an algorithm and the development of an artefact inspired by the investigation. Students were also asked to develop their ability to critique work – both their own and others' – by providing a self-assessment of their own artefacts, as well as, critically examining the submission of another student's assignment to coursework 1.

General remarks

At level 5 examiners expect students to be able to present coherent academic writing alongside precise citation and referencing. Insightful analysis and discussion of creative artefacts is expected and students should demonstrate how they are developing their technical abilities.

It is important to note that many students needlessly lost marks by not following the instructions specifying the appropriate format for submission. These instructions are not given arbitrarily but serve a specific purpose, i.e., making the task of marking more straightforward for the examiners. If examiners spend extra time searching through files that have not been labelled correctly it detracts from the work overall and marks are therefore deducted.

Coursework assignment 1: Distance metrics

There were five parts to this coursework assignment and many students made a reasonable attempt at all five. Unfortunately, about a quarter of students lost marks by formatting their submissions incorrectly and around a fifth of students did not submit anything for part D, thereby losing out on all possible marks for this section. This was a shame, especially in one case where a student submitted excellent work for the first three parts but then obtained zero for the fourth, bringing their mark down significantly. It is always better to submit a response for each section, even if partial or incomplete, rather than to completely omit any part.

Part A

This section required a description and discussion of the Levenshtein algorithm followed by a discussion of its potential applications to music. Students were expected to read appropriate books and articles as well as consider the musical context. Particularly good answers included: clear descriptions of the algorithm itself, potential extensions (such as the Hamming distance) and adept discussions of musical applications. One student discussed the use of edit distance to compare the similarities of rhythm in music, as well as, melodic similarities. This student buttressed their answer by including appropriate citations for their work. Not all students seemed to grasp that a very important aspect of applying the Levenshtein algorithm to specific cases is converting the information to something encoded as a string, only the stronger submissions covered this well.

Part B

This section required the implementation of a Processing sketch to determine the distance between a pair of strings. A number of students developed sketches that required the input of two words, which was not asked for in the brief. One excellent submission noted that for country names with high similarity, such as Togo and Tonga, when given an input that is the equal distance to both (e.g. 'Togna') the algorithm may choose the less intuitive option if alphabetic order is used. The student suggested returning all options with the same distance and allowing the user to choose the preferred one.

Some students restricted their algorithms to handle an edit distance of only one. This is not a reasonable assumption and in most cases this choice was not justified. One student included a lot of material about the complexity calculations of recursive algorithms, which was both incorrect and not relevant to the coursework. It is essential to check that the material is correct and take particular care in order to avoid including extraneous material.

Part C

This section required creative development from the earlier work and some students showed great initiative. A particularly commendable example used the Levenshtein distance as the basis for a typing game meant to test and improve typing speed. Another very creative sketch, also based on typing, produced different images based on the proportion of correctly spelt words entered. The sketch's use of colour and shape was imaginative. An intriguing and impressive idea was to use names and produce a visual artefact that was constrained and shaped in terms of the Levenshtein distance of a name input in relation to a list of names that had been previously stored. Grayscale and colour intensity were parameters as well as shape and movement.

Part D

This section was generally weak across the board, often with shallow investigation and exhibiting little insight. However, numerous submissions gained some marks – far better than not including anything. There were some very insightful submissions and arguments, although a couple of students were penalised for poor academic practice concerning their citation and referencing. One strong idea was to use multiple sequence alignment to compare traits between songs. Another good response noted that both textual and acoustic similarities can be measured for

music (textual content being used as the meta-data), concluding that the acoustic comparison would be more useful for recommender systems. This submission was very well argued and justified with specific mention of Echo Nest which uses both aspects.

Part E

This section required students to identify and provide information on a partner's submission, all but one student obtained the full 2%, which is a significant improvement on previous years.

In general, the coursework was approached well. Most students included reasonable comments in their code. However, in a few cases, some work was weakly coded and not commented on at all.

Coursework assignment 2: Inbetweening

The second coursework assignment began with a requirement to critique a partner's work and gave students an opportunity to examine another algorithm in depth. Fewer students omitted entire sections than in the first coursework. As a result, very few students (less than 10%) did not obtain a pass mark. We were pleased with the many excellent submissions, with around a third of students obtaining a first class mark.

Part A

The critique was done very well by some students, with one student obtaining full marks and another obtaining 19 out of a possible 20. Note that these submissions are available in the 'Examples of Excellent Work' section in the CO2227 course area on the VLE. Good submissions included insightful commentary on both the technical and creative aspects of their partner's work. Weaker submissions suffered from the following: comparing the work being critiqued with the student's own submission, giving a straightforward average of marks for each section to obtain a total (instead of a weighted average as instructed), focusing only on the essay and the technical aspects of the implementation, and not considering the creative aspect of the artefact or its impact. Some students also failed to comment on the submission's own critique where included. In contrast, good critiques featured commentary on the quality of code as well as the discussion itself, included a self-evaluation in the submission and made relevant suggestions for improvements.

Part B

In Part B students were required to present an academically appropriate description of tweening and its role in animation in relation to algorithmic approaches to animation. Some submissions contained reasonable academic content but were penalised for poor academic practice on citation and referencing. Other work exhibited a strong demonstration of conceptual understanding but suffered from an excessively narrow focus. Very good submissions were coherent and academically sound with appropriate justification for the assertions that they made.

Part C

Part C produced some nice work, some of it more technically focused but of a high standard. The better submissions were strong across the different implementations with good analysis and discussion. Weaker discussions contained evaluations of the students' own code rather than of the results or performance of the different algorithms. Other weaknesses included not

describing the criteria for comparison or choosing inappropriate criteria, such as code features. One student used a Venn diagram for part of their comparison, which was a good idea, but then did not follow through with a detailed evaluation.

Part D

Part D also included some very strong submissions. Some of these were based on simple ideas which can often be very effective. It is important to produce work that makes an impact as well as being technically strong and some students managed to do this to a very high degree. However, a number of submissions included very limited development of the ideas taken from part C and suffered from a lack of creativity and imagination. Another point of weakness was when a good idea was proposed but relied on a large proportion of code taken from other sources (even where acknowledged), this meant fewer marks were obtained for originality.