Final Project Proposal

Goals

Using as many concepts covered in ITP 116 as possible, design a Python program that allows students to take a mock exam of MATH 425a (see description below). Following a certain rubric, the program will automatically calculate the score based on the student's inputs. Of course, the program is not able to automatically grade free response questions, so all questions will be multiple choice or fill-in-the-blank.

MATH 425a, *Fundamental Concepts of Analysis*?

- Considered by many math majors as the "most notorious 400-level math course" at USC.

- *In the world of mathematics, our human intuition can sometimes lead us astray, and many pathologies and exotica arise if we are not extremely careful with our mathematical assumptions.* MATH 425a attempts to address these issue by reinventing mathematics from scratch (axioms) and see where — and how — can things sometimes go wrong.

- A proof-based class, 425a is unfriendly to students without prior exposure to proofs, thereby making it one of the more challenging undergraduate math courses.

- *It is said that when you see a 425a problem, you either think "this needs a proof?" (for example, formally prove $-(-x) = x$) or you think "this can be proven??"*

- Fun fact: one section of 425a is currently (fall 2021) using my notes as their "textbook". See here. Sadly I didn't get paid. Lol.

Requirements

(1) Regarding exam format and grading:

   (i) There are 20 questions in total and each question is worth 5 points.

   (ii) For the first part (T/F questions), there is a "question bank" consisting of 20 questions. 10 among them will be chosen randomly and put on exam. At the beginning of the exam, the program should tell the student which ten questions have been picked. The question numbers will remain unchanged.

   (iii) The second part consists of fill-in-the-blank questions. Since students may have slightly different answers, a similarity metric will be used (see here) when computing the grade. That is, if the student's answer is sufficiently similar to the correct one, it will be considered as correct. (I have designed the problems in a way that ensures the student can only produce a similar outcome when they actually solve the problem correctly.)

(2) The program will begin by telling the student that they are about to take a 425a exam. The timer will start once the student is done inputting their name and student ID as well as signing an Honor Pledge.

(3)   The program will provide the student with an URL link to a PDF version of the test (*this is because math formulas are much more easily typesetted in LaTeX than in Python, so I instead created a PDF version of the test using LaTeX.*) The program will also tell the student which questions need to be answered (see (1)(ii)). The PDF file has been attached to the end of this Proposal.

(4)   Unless the time limit has been reached, the program will always ask for the student's input. (Also, give appropriate prompts so the student knows what to type.)

   (i)   If the student types 'time', the program will output the remaining time of the exam.

   (ii) If the student types a number that corresponds to a question appearing on the test, ask the student for another input which will be stored as the student's answer to that question. If the student enters 'back', abort reading input.

   (iii) If the student types something else, inform the student that input is invalid and ask for input again.

(5)   If a question has been previously answered and the student provides a different answer, the program will ask for the student's confirmation that they want to replace the previous answer.

(6)   (Optional) The exam is 50 minutes long; the program will notify the student when there are 30 minutes, 10 minutes, 5 minutes, and 2 minutes left, respectively. *For practical reasons, in this program, replace the total length by 3 minutes, and remind the students when there are 1 minutes, 30 seconds, and 10 seconds left, respectively.* You may use codes from here, for example.

(7)   When time runs out, interrupt any input (see (12)(iv) below) and start calculating the student's raw score.

(8)   Add a 2-second delay using `time.sleep(2)` when calculating the score to make it look more realistic.

(9)   The program will also generate a `.txt` file and save the student's answers in that file. The file name should contain student's name and/or other info.

(10) Finally, the program will display the computed raw score, including a detailed report for each question. Give the student an estimated curve and a corresponding letter grade before exiting the program.

(11) The program should make use of topics covered in lectures. It must use functions and it must make use of all of the following:

$$\{\text{Boolean expressions, If statements, For/while loops, Strings, Lists/dictionaries, File I/O}\}$$

(12) The program should define (and use) multiple functions, including `main()`. Among these:

   (i)   at least 4 functions should each take at least one parameter;

   (ii) at least 2 functions should each return at least one value;

   (iii) at least 5 functions should not involve `print()`; and

   (iv) define a timer function that automatically stops recording any input when time runs out (since this is beyond the scope, for this function, online codes such as this are allowed). *If the student is trying to type something in the command line but didn't manage to type Enter, discard the input to mimic the situation where the student only has answers on their scratch paper but not the answer sheet.*

Sample Output

Deliverables and Submission Instructions

- Create a folder with name of form **ITP116_A3_LastName_FirstName**.

- Inside the folder, put your Python source code.

- Compress the folder: on MacOS, select the file - right click - compress.

- Upload the zip file to your Blackboard section:

  - On Blackboard, click on the Assignments item in the course menu on the left.

  - Click on the specific item for this assignment.

  - Click on the "Browse My Computer" button and select your zip file.

  - Click the Submit button.

Grading Criteria

(a) 1.0% Documentation and comments are clear and descriptive.

(b) 1.0% Requirements (10)(i) is met; 0.25% for each function.

(c) 0.5% Requirements (10)(ii) is met; 0.25% for each function. (Can be functions appearing in the previous criterion.)

(d) 1.0% Requirements (10)(iii) is met; 0.2% for each function. (Can be functions appearing in previous criteria.)

(e) 1.0% Requirements (10)(iv) is met. That is, the timer works properly.

(f) 8.0% The program runs without logic error. That is, the student can use all the features (pick question, answer question, ask for remaining time, pick question but then unpick it,) as described in the Requirements section. 2.0% for each feature.

(g) 2.0% The program correctly sets up the T/F section by choosing 10 different questions from the "question bank".

(h) 2.0% The program correctly calculates the student's score.

(i) 1.5% The program correctly outputs the detailed score report.

MATH 425a Fake Exam

November 25, 2021

True or False Questions

True or false questions; for each question, please either type 'T' or 'F' in Python. For a statement that is true under some circumstances but false in general, answer 'F'. On the test, **ten** questions will be chosen randomly.

(1) All dense subsets of $\mathbb{R}$ are countable.

(2) The collection of finite length binary strings is countable, whereas the collection of infinite length binary strings is uncountable.

(3) Every norm induces a metric, and every metric induces a norm.

(4) Under the notions of addition and multiplication, $\mathbb{N}, \mathbb{Z}, \mathbb{Q}$ are not fields but $\mathbb{R}$ and $\mathbb{C}$ are.

(5) An arbitrary union of closed sets is closed.

(6) The only compact subsets of $\mathbb{R}$ are intervals of form $[a, b]$, where $a < b$ and both are finite.

(7) Fix a rational number q. Then
$$\limsup_{n \to \infty} (nq - \lfloor nq \rfloor) < 1.$$

(8) Fix an irrational number r. Then
$$\limsup_{n \to \infty} (rn - \lfloor rn \rfloor) = 1.$$

(9) Let $\{a_n\}$ be a sequence. If $\limsup_{n \to \infty} |a_{n+1}/a_n| \leqslant 1$ then $\sum a_n$ converges absolutely.

(10) A closed and bounded subset of $\mathbb{R}^n$ is compact. (Here we are using the standard Euclidean metric.)

(11) In a complete metric space, a set is compact if and only if it is bounded.

(12) The continuous image of a open set is open.

(13) The continuous image of a compact set is compact.

(14) The continuous image of a connected set is connected.

(15) The continuous image of a disconnected set is disconnected.

(16) If f is differentiable at x and $f'(x) > 0$, then f is locally increasing.

(17) If a function is Riemann-Stieltjes integrable then it is Riemann integrable.

(18) Uniform convergence is equivalent to convergent in sup norm.

(19) If $f_n \to f$ pointwise, and f_n, f are all continuous, then $f_n \to f$ uniformly.

(20) If a function is continuous, it must be differentiable at at least one point.

Fill-in-the-blank Questions

Try to fill in the blanks as accurately as possible. The grades will be computed automatically, but rooms for typos will be given: for e.g., if the correct answer is "`finite-dimensional`" and you wrote "`finite demensional`", you will still receive full credit, but if you typed "`banana%#&`" you will of course not receive any points!

The answer for each question consists of *exactly* one word. One of them includes a hyphen. All letters should be in lowercase.

(21) A function is continuous if the _____ (noun) of any closed set is closed.

(22) A set E is _____ (adjective) if it is closed and has no isolated point; that is, every point in E is a limit point.

(23) The limsup of a sequence is the supremum of all subsequential _____ (noun, plural).

(24) A sequence _____ (verb) if and only if its limsup equals its liminf.

(25) A sequence that converges but diverges absolutely is said to converge _____ (adverb).

(26) If f_n converges to f _____ (adverb) and each f_n is continuous, then f is continuous.

(27) The Weierstraß Approximation Theorem states that (Bernstein) _____ (noun, plural) are dense in $C([0,1])$.

(28) Banach Contraction Principle: in a Banach space, a strong contraction has a unique _____ (noun).

(29) The Devil's staircase / Cantor function is differentiable almost everywhere; wherever the derivative exists, it equals _____ (a English word).

(30) The Weierstraß Monster is _____ (answer something other than '`not`') differentiable.