LATE

minutes remaining

Hide Time

Manual Save

Navigation

Question 1. Word Bank Matching (1 point each, 14 points)

For each statement below, input the letter of the term that is *best* described. Note that you can click each word (cell) to mark it off. Each word is used at most once.

A. — Adapter Design Pattern	B. — Bug Bounties	C. — Concurrency Bug	D. — Delta Debugging
E. — Fault Localization	F. — Functional Requirements	G. — Fuzz Testing	H. — Informal Goal
I. — Mocking	J. — Multi-Language Projects	K. — Mutation Testing	L. — Named Constructor Idiom Pattern
M. — Perverse Incentive	N. — Productivity	O. — Profiling	P. — Quality Requirement
Q. — Readability	R. — Requirement Elicitation	S. — Risk	T. — Singleton Design Pattern
U. — Stakeholder	V. — Traceability	W. — Triage	X. — Validation
Y. — Weak Conflict			

Q1.1: Whenever software engineers at MoonChips try to push new code into the codebase, their code is automatically checked by linters and style checkers to help ensure that future engineers can more easily understand it, with the goal of improving **THIS VOCAB TERM**

Q1.2: Some users of Ann Arbor Public Library's website are reporting issues where they attempt to place a book on hold that the website says is available. However, when they click 'Confirm', they are shown a 'Book Not In Stock' error. The website's developers theorize that this is due to different members placing the same book on hold at the same time and hope to use CHESS to help identify this kind of bug.

Q1.3: PiedPiper is building airplane control software. They switch their development methodology from an agile model to a spiral model hoping that it will help mitigate **THIS VOCAB TERM**

Q1.4: Adit is working on a new music streaming application. Instead of using various global variables to track certain song information, they track these properties using only one instance of a class.

Q1.5: Momo works for Sleddit, a new image sharing service. They have 1000 pictures that they use as test inputs, which leads to 97% code coverage. They believe this amount of videos is excessive and want to find the minimal set of videos that will have the same coverage.

Q1.6: In order to take advantage of the new promotion system at Veecsa where employees' chance at a promotion is tied to the amount of code they push to the codebase, Jingyi starts writing excessively verbose code, even at the cost of readability.

Q1.7: Arjav wants to write a test for the purchase confirmation screen on Rulu's new ecommerce platform. However, they don't want to purchase an actual item. Thus, they create a fake product and API response to avoid this.

Q1.8: MunchyRoll has a very small engineering team that has been working on a banking app. As a result, they may have missed some subtle bugs that could compromise their customers' security. They start a program where users can report novel bugs in their software in exchange for a monetary reward

Q1.9: Miscord was hired by a grocery store to make an app for them. Before Miscord's development team starts coding, they meet with the store's management to perform **THIS VOCAB TERM**, with the goal of learning about the requirements, objectives, and assumptions regarding the app.

Q1.10: Antony is working on a music streaming app. They want the final product to start playing music after the user pushes the ‘play' button with a maximum latency of 3 seconds.

LATE

minutes remaining

Hide Time

Manual Save

Navigation

Q1.11: MVidia is in the ideation phase of a new online movie rental platform. One stakeholder provides the requirement that renters should lose access to a movie after 30 days. Another stakeholder says that renters should lose access to a movie only after the first time that they watch it.

Q1.12: It is taking customers way too long to render the cart screen on EECSon Mobile's online shopping platform. Sandy uses a tool to see which parts of the code are taking a long time to run, where they notice that most of the execution time is being spent in the function that retrieves the customer's forms of payment, so they try to optimize it.

Q1.13: Emily made a bunch of commits to a repository and now tests are failing! They are not sure which lines are causing the error so they use git bisect to identify which specific commit introduced the error.

Q1.14: Sukuna is working for Wahoo and decides to make a bug report page to track all the reported issues that have been popping up for their latest product. Now, Sukuna is figuring out how to decide and assign a priority to each of the listed bugs - this process is known as **THIS VOCAB TERM.**

Question 2. Delta Debugging (21 points)

```
1    [0, 1, 2, 3, 4] – Not Interesting
2    [5, 6, 7, 8, 9] – Not Interesting
3    [0, 1, 2, 3, 4, 5, 6] – Not Interesting
4    [0, 1, 2, 3, 4, 7, 8, 9] – Not Interesting
5    [0, 1, 2, 3, 4, 5, 6, 7] – Not Interesting
6    [0, 1, 2, 3, 4, 5, 6, 8, 9] – Interesting
7    [0, 1, 2, 3, 4, 5, 6, 8] – Not Interesting
8    [0, 1, 2, 3, 4, 5, 6, 9] – Not Interesting
9    [0, 1, 2, 3, 4, 7, 8, 9, 5] – Interesting
10   [5, 6, 7, 8, 9, 0, 1] – Not Interesting
11   [5, 6, 7, 8, 9, 2, 3, 4] – Not Interesting
12   [5, 6, 7, 8, 9, 0, 1, 2] – Interesting
13   [5, 6, 7, 8, 9, 2, 3, 4, 0] – Not Interesting
14   [5, 6, 7, 8, 9, 2, 3, 4, 1] – Interesting
15
```

(a) (3 points)

What's The Answer Given The Logs?

Consider running the delta debugging algorithm (from EECS 481) on input [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]. The calls to `Interesting()` are shown above along with their results. Based on this, please determine the minimal interesting subset that the algorithm will return.

Please format your answer in the form [x, y, z, ...]; for example, [0, 1, 2].

Your answer here.

(b) (6 points)

Fill In The Blanks

The following table displays "True" or "False" for the **first three** `Interesting()` calls. This question involves an `Interesting()` function that's independent of the one from part A. Given the answers for three of these calls, fill in the expected output for the rest of the calls in this table.

If True, please also indicate which property (Ambiguity, Monotonicity, or Consistency) allows you to conclude that the subset is interesting.

If False, please also indicate which property (Ambiguity, Monotonicity, or Consistency) would be violated if the subset was interesting.

Subset	Output
Interesting({5, 9})	True
Interesting({5, 8})	True
Interesting({9})	False
Interesting({1, 2, 5, 7, 9})	Question 2.b.i

Subset	Output
Interesting({5})	Question 2.b.ii
Interesting({7, 9})	Question 2.b.iii

(b.i) (2 points)

- ○ A) True (Ambiguity)
- ○ B) True (Monotonicity)
- ○ C) True (Consistency)
- ○ D) False (Ambiguity)
- ○ E) False (Monotonicity)
- ○ F) False (Consistency)

(b.ii) (2 points)

- ○ A) True (Ambiguity)
- ○ B) True (Monotonicity)
- ○ C) True (Consistency)
- ○ D) False (Ambiguity)
- ○ E) False (Monotonicity)
- ○ F) False (Consistency)

(b.iii) (2 points)

- ○ A) True (Ambiguity)
- ○ B) True (Monotonicity)
- ○ C) True (Consistency)
- ○ D) False (Ambiguity)
- ○ E) False (Monotonicity)
- ○ F) False (Consistency)

(c) (3 points)

DD Performance

Please indicate if the following statement is True or False, **and** include a brief justification. Limit your entire answer to at most 2 sentences.

Delta Debugging is most efficient when a single change is causing the failure

Your answer here.

(d) (9 points)

Will Delta Debugging Be Useful Or Not?

Please indicate whether or not Delta Debugging would be useful in each of the following scenarios. If "Not Useful", please also indicate the reason why.

(d.i) (3 points) You write a large test suite in order to get 100% line coverage when testing your source code. After coding for hours, you notice that your tests are starting to get repetitive and you suspect that your suite might already be sufficient so you want to remove the redundant tests.

- ○ A) Useful
- ○ B) Not Useful (Inconsistent)
- ○ C) Not Useful (Ambiguous)
- ○ D) Not Useful (Not Monotonic)

(d.ii) (3 points) We have the following line of HTML that fails to pass the HTML parser. We want to find which specific attribute is causing the issue.

```
“<input className=”textInput” name="color" holder="Enter for favorite color..."  type="text" value={this.state.color} onChange={this.handleChange} required /> ”
```

- ○ A) Useful
- ○ B) Not Useful (Inconsistent)

LATE

minutes remaining

Hide Time

Manual Save

Navigation

C) Not Useful (Ambiguous)
D) Not Useful (Not Monotonic)

(d.iii) (3 points) You have a list of 10 numbers that are greater than 0, and you want to employ the delta debugging algorithm to find a minimal subset of those numbers that will add up to x. There are multiple subsets of numbers whose sum will add up to x, but you don't know what the subsets are.

A) Useful
B) Not Useful (Inconsistent)
C) Not Useful (Ambiguous)
D) Not Useful (Not Monotonic)

Question 3. Short Answer (20 points)

(a) (5 points)

You've been tasked with debugging a large multi-threaded web server application that frequently crashes under heavy load. Users report intermittent timeouts and unresponsive behavior during peak usage periods. As part of the debugging process, you need to identify and resolve any threading issues that may be causing these performance problems.

Should you use static or dynamic analysis? Explain your reasoning.
Please limit your entire answer to no more than 3 sentences.

Your answer here.

(b) (5 points)

You're currently working on a design document which details the design of the work of a new, upcoming project that you will be working on. After you finish writing the design document, you show it to your manager for approval. However, halfway through reading the design document, your manager strongly disagrees with the tech stack you propose to use for the project. Unfortunately, you fail to see why your manager disagrees, and you believe your original approach is better.

Trying to de-escalate the conflict as much as possible, you look to use your knowledge from EECS 481. What are **two** methods you can use in this scenario to help resolve the conflict? For each method, explain why you can use it to help de-escalate the conflict.
Please limit your entire answer to no more than 4 sentences.

Your answer here.

(c) (5 points)

You're part of a software development team working on a legacy codebase for a large e-commerce platform. The codebase consists of numerous classes with similar functions scattered across different modules, leading to poor readability and maintainability. For instance, there are multiple classes handling product inventory management, each with its own set of methods for adding, updating, and deleting products. The lack of consistency and organization makes it challenging for developers to understand and modify the code efficiently.

Please list **two** concepts or techniques from EECS 481 that you can use, in order to make the codebase more maintainable? Please illustrate them, and explain your reasoning.
Please limit your entire answer to no more than 4 sentences.

Your answer here.

(d) (5 points)

In Professor Kochunas' lecture about "Software Engineering Practices in Scientific Computing", he discusses quality assurance and testing.

LATE

minutes remaining

Hide Time

Manual Save

Navigation

Imagine you're part of a software development team tasked with creating a new mobile banking application for a large financial institution. The application aims to provide users with seamless banking experiences, including account management, fund transfers, bill payments, and financial insights. The project timeline is tight, with a deadline set in three months. The application needs to be secure, user-friendly, and fully functional across various devices and operating systems. Additionally, the application must comply with strict financial regulations to ensure the safety and privacy of users' financial data.

What are **two** quality assurance activities that you should perform while creating this application. Please also explain your reasoning.
Limit your entire answer to no more than 5 sentences.

Your answer here.

Question 4. Fault Localization (15 points)

Consider the following Python function snippet `song_statistics()`. It takes in a song information dictionary as input. The dictionary maps song attributes to lists containing information about that particular attribute. The current implementation calculates the number of songs with a particular length, in addition to generating playlists and other (hidden) functionality.

```
1. def song_statistics(song_info: dict):
2.     song_types = {'short-song': 0, 'long-song': 0, 'medium-song': 0}
3.
4.     for length in song_info['lengths']:
5.         if length > 300:
6.             song_types['long-song'] += 1
7.         elif length < 100:
8.             song_types['short-song'] += 1
9.         else:
10.            song_types['medium-song'] += 1
11.
12.    if song_types['long-song'] > 10:
13.         generate_playlist()
14.
15.    while len(song_info['titles']) > 100:
...         ...
21.    print('song statistics captured!')
```

Consider the following table. Each row corresponds to one test case execution. In particular, each row reports one run of the program, in which it takes some song info (not shown) as input and produces output that is either correct ("Pass") or incorrect ("Fail"). Each row also includes the lines visited while the program executes on that input.

Test Case	Status	Lines Visited
1	Pass	[2, 4, 5, 6, 12, 15, 16, 17, 18, 19]
2	Fail	[2, 4, 12, 13, 15, 16, 17, 18, 19]
3	Pass	[2, 4, 12, 13, 15, 16, 17, 18, 19]
4	Pass	[2, 4, 5, 6, 19, 21]

(a) (5 points)
Using the table information above, compute the Tarantula suspiciousness score for the top 3 most suspicious lines shown in `song_statistics()`, and provide these scores along with the line number in the answer box below. Express the final answer as a list of tuples of line numbers (ints) and scores (floats, rounded to 2 decimal places), sorted by score descending and then by line number ascending; for example, if line 2 has a suspiciousness of 1, line 1 has a suspiciousness of 0.5, and line 5 has a suspiciousness of 0.5, your answer should be `[(2, 1.00), (1, 0.50), (5, 0.50)]` .

Your answer here.

LATE

minutes remaining

Hide Time

Manual Save

Navigation

(b) (3 points)

Which line from the above list of lines, do you think, is causing the problem and why? If you think the list does not contain a relevant line, please indicate that and explain why. Limit your entire answer to at most 3 sentences.

Your answer here.

(c) (5 points)

As a developer on a software project, you're currently running into an odd bug, and you decide to find the bug using fault localization techniques learned from EECS 481. However, you're unsure whether you should proceed with a fault localization tool or not. Using no more than 4 sentences, compare and contrast (1) tool-based fault localization and (2) manually performing fault location.

Your answer here.

(d) (2 points)

In addition to fault localization techniques, you look to use profiling to aid in understanding your program's behavior better. Give one example of when you could use profiling to understand your program better and how it could help in identifying and fixing errors. Use at most 2 sentences.

Your answer here.

Question 5. Design Patterns (15 points)

Suppose you are developing a tax system, and have written the following code.

```
def calculateTax(subtotal, country, state, taxrates):
    taxrate = 0
    if country == "US":
        taxrate = taxrates[state]
    else:
        taxrate = taxrates[country]
    return subtotal + subtotal * taxrate

def findTimeZone(country, state, zones):
    timezone = None
    if country == "US":
        timezone = zones[state]
    else:
        timezone = zones[country]
    return timezone
```

(a.i) (1 points)

During the code review session, your mentor revised your code to the following.

```
def USvsNonUSLookup(country, state, lookup):
    foundValue = None
    if country == "US":
        foundValue = lookup[state]
    else:
        foundValue = lookup[country]
    return foundValue
```

```
def calculateTax(subtotal, country, state, taxrates):
    taxrate = USvsNonUSLookup(country, state, taxrates)
    return subtotal + subtotal * taxrate


def findTimeZone(country, state, zones):
    timezone = USvsNonUSLookup(country, state, zones)
    return timezone

```

LATE

minutes remaining

Hide Time

Manual Save

Navigation

What is an issue that your mentor was trying to address?

- ◯ A) Copy-pasting code of the same functionality
- ◯ B) Confusing variable naming
- ◯ C) Bad indentation and format
- ◯ D) Excessive use of global variables

(a.ii) (2 points)

Please describe two drawbacks of confusing variable naming. Limit your entire answer to at most two sentences.

Your answer here.

(b.i) (2 points)

This design pattern ensures a class only has one instance, and provides a global point of access to it.

Please choose the design pattern that best matches the description above.

- ◯ A) Iterator
- ◯ B) Proxy
- ◯ C) Observer
- ◯ D) Singleton

(b.ii) (2 points)

You have an existing unchangeable code base which uses `player` (an instance of `MP3Player`) to play the audio. You now want to additionally support playing the audio of a `MP4File` by utilizing its provided `play_music`, without changing the codebase. You wrap it in a new class `MP4AudioPlayer` inherited from `MP3Player`, and implement the compatible interface play.

```
def existingUnchangeableCodeBase(player):
    player.play()


class MP3Player:
    def play(self):
        print(f"Playing MP3 file.")


class MP4File:
    def play_music(self):
        print("Playing audio of a MP4 file")


class MP4AudioPlayer(MP3Player):
    def __init__(self, mp4_file):
        self.mp4_file = mp4_file

    def play(self):
        self.mp4_file.play_music()

```

Please choose the design pattern that best matches the description above.

- ◯ A) Observer
- ◯ B) Singleton
- ◯ C) Iterator
- ◯ D) Adapter

LATE

minutes remaining

Hide Time

Manual Save

Navigation

(b.iii) (2 points)
Consider the following description for a design pattern X. X provides an interface for creating objects, but allows subclasses to decide which class to instantiate. X is useful when you want to abstract the object creation process and create objects without specifying the exact class.

Please choose the design pattern that best matches the description above.

- ○ A) Singleton
- ○ B) Adapter
- ○ C) Iterator
- ○ D) Factory

(b.iv) (2 points) The design pattern provides a uniform interface for traversing containers (eg. vectors, list-like objects) regardless of how they are implemented.

Please choose the design pattern that best matches the description above.

- ○ A) Iterator
- ○ B) Adapter
- ○ C) Singleton
- ○ D) Proxy

(c) (2 points)
Suppose that a particular software development project spends X=38% of its lifetime effort on implementation, Y=45% of its lifetime effort on testing, and Z=17% of its lifetime effort on other non-testing maintenance. You have proposed a new design, and you would like to evaluate its effectiveness. In particular, you have already concluded:
(a) this new design would increase the effort required for implementation by M=15% (for example, if implementation previously took 10 hours, with an increase effort by 35%, it would now take 13.5 hours);
(b) but this new design would also reduce the effort required for testing by N=29%.

Assume the project originally required 100 hours to complete. Now, with this new design, please calculate the hours required for the same project. **Round your answer to the nearest integer.** For example, 3.4 would be rounded to 3, and 3.6 would be rounded to 4.

Your answer here.

(d) (2 points)
Support or refute the claim: Given that the time required to read code during activities like code reviews or inspections is proportional to the number of lines, and understanding code is crucial in software maintenance, writing accurate programs with minimal lines of code emerges as the optimal design approach for maintenance. This strategy not only saves time but also enhances the maintainability of software by simplifying comprehension and debugging processes.
Please use concrete lecture and/or reading materials, to back up your answer. Limit your entire answer to at most 4 sentences.

Your answer here.

Question 6. Requirements Elicitation (8 points)

Suppose you are tasked to develop a mobile banking application for a new bank. This application should be capable of making transactions internationally, and catering to both individuals and business clients.

(a) (2 points) List 2 possible **functional** requirements for the application. Each requirement should be described using 1 sentence. Limit your entire answer to at most 2 sentences.

Your answer here.

(b) (2 points) List 2 possible **non-functional** requirements for the application. Each requirement should be described using 1 sentence. Limit your entire answer to at most 2 sentences.

Your answer here.

(c) (4 points) Identify 2 possible stakeholders from the application, and describe a conflict that might arise between the 2 stakeholders. Explain **whether** the conflict you mentioned is a strong or weak conflict and **why**. Limit your answer to no more than 4 sentences.

Your answer here.

Question 7. Interview (7 points)

Bob is tasked with interviewing an applicant Alice for a software development position and evaluating Alice's technical skills.

The interview question is: given an integer array `nums` and an integer `val`, remove all occurrences of `val` from `nums`. The length of `nums` can be zero or larger. Return the length of the resulting array.

```python
def removeElement(nums: List[int], val: int) -> int:
    index = 1   # point to the end of the array after element removal
    for i in range(1, len(nums)):
        if nums[i] != val:
            nums[index] = nums[i]
            index += 1
    return index

```

(a) (2 points) Upon receiving the problem, Alice asked about the time complexity requirement, and whether the removal should happen in-place. After a short while, Alice presented the above code to you.

Identify **two** points where Alice did well during the interview.

Your answer here.

(b) (5 points) Suppose you are Alice. After you delivered the solution above, Bob prompted you that there was a bug in your code. Bob also asked you to write test cases to better test the code (at least to reveal the bug).
In your answer, please:
(1) describe the bug in one sentence;
(2) give one test case that can expose the bug;
(3) provide an additional test case, different from the one in (2), that you believe is also necessary for better testing the code, and explain the rationale behind it.
Please limit your entire answer to at most five sentences.

Your answer here.

Extra Credit

LATE

minutes remaining

Hide Time

Manual Save

Navigation

(1) What was your favorite topic or activity during the course? (1 point)

Your answer here.

(2) What is one thing that you think we should do more of next semester? (1 point)

Your answer here.

(3) What is one thing you would most recommend that we change for future semesters? (1 point)

Your answer here.

(4) What're your thoughts on adding some more advanced lectures that dive deeper into a couple topics (such as testing, static analysis, model checking, program synthesis) in more depth? (1 point)

Your answer here.

(5) Identify a single optional reading that was assigned after Exam 1. Write two sentences about it that convince us you read it critically. Please make sure that you identify the title of the reading. (2 points)

Your answer here.

Honor Pledge and Exam Submission

You must check the boxes below before you can submit your exam.

- [ ] I have neither given nor received unauthorized aid on this exam.
- [ ] *I am ready to submit my exam.*

Note that your submission will be marked as late. You can still submit, and we will retain all submissions you make, but unless you have a documented extenuating circumstance, we will not consider this submission.

Submit My Exam

Once you submit, you will be able to leave the page without issue. Please don't try to mash the button.

The exam is graded out of 100 points.