LATE

minutes remaining

Hide Time

Manual Save

Navigation

- Question 1
- Question 2
- Question 3
- Question 4
- Question 5
- Question 6
- Question 7
- Question 8
- Extra Credit
- Pledge & Submit

General Reminders

- You **may** use *free* generative AI tools on this exam.
- The exam is open note, open lecture recordings, open course webpage, open computer (including Visual Studio, Python, etc.), and open Internet in general. You may use general webpages (e.g., Stack Overflow, Wikipedia, etc.).
- You may **not** use live help from another human.
- If you leave a sub-question entirely **blank** then you will receive one third of its points rounded down. For example, if a sub-question is marked "(4 points)", you would receive 1 point if you left it entirely blank.
- If you are asked to support or refute a position, indicate which option you selected. One word can suffice.
- If you are asked to provide a quote from multiple possible sources (e.g., either the lecture slides or a reading), indicate which source you selected.
- If you quote from a lecture slide, you must indicate which slide number. If you quote from something spoken verbally (out loud) during a lecture, you must include a timestamp.
- If you are asked to provide a quote to help support or justify your argument, your quote must be directly relevant to your argument.
- If we cannot verify that your quote is from one of the required sources (or if it appears to be fabricated), you will receive **no points** for that entire question.

Question 1. Word Bank Matching (1 point each, 15 points total)

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

A. — Causation	B. — Complexity	C. — Composite Design Pattern	D. — Correlation
E. — Creational Design Pattern	F. — Delta Debugging	G. — Designation Conflict	H. — Elicitation
I. — Fault Localization	J. — Informal Goal	K. — Large Language Model	L. — Maintainability
M. — Productivity	N. — Profiling	O. — Program Synthesis	P. — Risk
Q. — Stakeholders	R. — Static Analysis	S. — Validation	T. — Weak Conflict

Q1.1: ☐ A descriptive model of software readability may include this type of relationship between blank lines and readability.

Q1.2: ☐ Iroh asks a tool, such as ChatGPT or GitHub Copilot, to generate source code. The code may be reviewed by humans, but it was created by an algorithm.

Q1.3: ☐ Ozai is directed to reduce memory consumption in a program. Ozai pursues this by instrumenting each method to determine how much memory it allocates at run time, and second by analyzing the results after running the program.

Q1.4: ☐ Andrew is writing software for a microphone company. The company's market research suggests that customers want the audio to "not sound too breathy". This information, without elaboration, is provided to Andrew to help guide software development.

Q1.5: ☐ This planning factor includes both the odds of an event happening and also the consequences of that event happening.

Q1.6: ☐ GlazeBook programmers conclude that the new software they develop is likely to exist for some time. They structure their software to simplify subsequent changes, including both fixing bugs and also adding new features.

Q1.7: ☐ The University of Michigan is considering revamping Wolverine Access. Lawyers, representatives from human resources, professors, and students are all gathered to provide input.

Q1.8: ☐ Aang structures a class so that the normal constructor cannot be called and another method must be called instead. This allows type information to be hidden.

This sort of relationship between software factors and outcomes provides the most confidence in

LATE

minutes remaining

Hide Time

Manual Save

Navigation

- Question 1
- Question 2
- Question 3
- Question 4
- Question 5
- Question 6
- Question 7
- Question 8
- Extra Credit
- Pledge & Submit

Q1.9:	☐	predictions about the future. It is desired in normative models.
Q1.10:	☐	Gojo uses a series of interviews, walkthroughs and checklists to ensure that the requirements are complete and consistent.
Q1.11:	☐	A tool, such as a neural network transformer, that can create text in response to natural language prompts.
Q1.12:	☐	FloorMart's informal documentation uses the term "priority" for both operating system scheduler precedence and also for a queue data structure used in shortest path calculations. This confuses developers.
Q1.13:	☐	This attribute of software is commonly computed statically, but popular metrics associated with it may not align with human perceptions or judgments.
Q1.14:	☐	Aamir is writing a program that has access to video files and wants to use an analysis library that operates on sets of images. By structuring the code to treat individual frames and group of frames uniformly, maintainability is improved. From the code's perspective, the analysis library can now be applied to sets of frames.

Question 2. Guest Lectures and Concepts (16 points)

Reminder: While not all guest lecturers were able to make their PDF slides available, all of the lecture recordings are available at https://leccap.engin.umich.edu/leccap/site/1gz3vacgfuusnfw31vh and your notes summarizing and organizing the content of those guest lectures are available to you. If you include a spoken quote from a lecture, you **must include a timestamp** so that the graders can verify it.

(a) (4 points)

Consider the guest lecture by Zak Fry with a particular focus on Slide 16 of https://eecs481.org/lectures/ZFGuestLecture_public.pdf discussing exploiting cognitive biases.

Dr. Fry was targeting a particular type of defect. Using the terminology from class, would you expect these to be low, medium or high *severity* defects and why? (Use at most two sentences.)

Summarize *cognitive load* in one sentence, and then include a quote from one of our course readings or lecture slides to support or elaborate on it. Are Dr. Fry's results most relevant to Formal Code Inspection, Passaround Code Review, both, or neither? (Beyond the quote, use at most three sentences.)

Your answer here.

LATE

minutes remaining

Hide Time

Manual Save

Navigation

- Question 1
- Question 2
- Question 3
- Question 4
- Question 5
- Question 6
- Question 7
- Question 8
- Extra Credit
- Pledge & Submit

(b) (4 points)

Consider the guest lecture from Arthur Krieger and Maya Vijan. In particular, around 44 minutes of the recording, Maya Vijan mentioned "Being able to communicate what is going on in a technical problem to a non-technical audience".

On slide 7 of "Requirements, Validation and Risk" https://eecs481.org/lectures/se-14-valid.pdf we gave examples of stakeholders for the NASA Near Shoemaker mission. Identify one stakeholder in that diagram that would be categorized as "non-technical" in this context and explain why communication with that stakeholder would be critical. (Do not pick the asteroid. :-)) Use at most two sentences.

Choose a conflict type identified on slide 28 of "Requirements, Validation and Risk". Name the conflict type, make up an example of that conflict type from the particular domain Krieger and Vijan are working in (be specific; general answers will not be accepted), and indicate how a software engineer and a non-technical audience might end up with inconsistent interpretations. Use at most two sentences.

Your answer here.

(c) (4 points)

Consider Alex Collier's guest lecture and the story he told around slide 10 (https://eecs481.org/lectures/collier-guest.pdf) of a performance issue related to REDcap (a secure tool for storing medical data in a legally compliant way). He indicates that his final solution involved "Python multithreading for concurrent processing". Support or refute the claim that a dynamic analysis like Delta Debugging (see slides 10 or 61 of https://eecs481.org/lectures/se-12-debug.pdf for how Delta Debugging might apply to thread schedules, etc.) or sampling-based profiling (see near slide 48 of https://eecs481.org/lectures/se-11-faultloc.pdf) would have helped get him in that trouble spot. Include a quote from the slides or readings (your choice, but indicate which one) to support your argument. (After copying the quote, use at most four sentences.)

Your answer here.

(d) (4 points)

Consider Henry Beckstein's guest lecture about working at Subaru Labs in Japan. At around the 62 minute mark in the recording (displayed as 1:02 in the UI), a student asks him a question about how he retrains himself or gains mastery in relevant concepts. Henry describes a technique ("Instead of looking at the problem …") that he has found effective.

Henry's answer closely aligns with one of the first five readings from our Productivity lecture (listed in the outline on slide 4 of https://eecs481.org/lectures/se-19-productive.pdf). Identify the reading and indicate how Henry's approach matches what we learned. Use a quote from Henry (exact spelling is not required, but include a rough timestamp for the graders; use the "CC" button to turn on captions if you are not certain) and also a quote from the Productivity slides or readings to support your answer. (After copying the two quotes, use at most three sentences.)

Your answer here.

Question 3. Delta Debugging (13 points)

Answer the following questions about the Delta Debugging algorithm.

(a) (5 points)

We discussed three key assumptions for the Delta Debugging algorithm: Monotonic, Unambiguous, and Consistent (https://eecs481.org/lectures/se-12-debug.pdf). Your company is tasked with finding a small Interesting subset of a customer dataset using Delta Debugging. For this customer's dataset and use case, Delta Debugging is Consistent. However, instead of mentioning Monotonic and Unambiguous, the customer reports that their dataset and notion of Interesting satisfy a Growth property:

LATE

minutes remaining

Hide Time

Manual Save

Navigation

- Question 1
- Question 2
- Question 3
- Question 4
- Question 5
- Question 6
- Question 7
- Question 8
- Extra Credit
- Pledge & Submit

- Forall X, Forall Y, Forall Z. Interesting(X) && Interesting(Z) → Interesting(Y ∪ (X ∩ Z))

Support or refute the claim that Delta Debugging will work correctly with the Consistent and Growth assumptions. Include a quote from the slides or readings (your choice, but indicate which one) to support your argument. (Beyond copying the quote, use at most four sentences.)

If your browser does not display the mathematical symbols correctly, here is another equivalent rendering of Growth: `Forall X, Forall Y, Forall Z. (Interesting(X) [AND] Interesting(Z)) [IMPLIES] Interesting(Y [UNION] (X [INTERSECTION] Z))` .

Your answer here.

(b) (3 points)

Consider the set suite of PNG files you used in Homework #5: https://eecs481.org/hw5/large-png-suite.zip. A customer wants you to find a small Interesting subset of that test suite, where **Interesting** is defined as "the total file size of all files in the subset is at least 8799 bytes". The total file size of a set of files is the sum of the individual sizes. Reminder: You can use or review your Delta Debugging implementation from class.

What is a minimal Interesting subset of that test suite? Format your answer as a Python list of strings. Example: `[ "1.png", "2.png" ]`

Scripting hint. You can use shell commands like "`cat … | wc -c`" to quickly determine the total file size of multiple files and shell operations like "`if [ x -ge y ]`" to quickly determine if x >= y. (You are not required to do so; this is just a reminder.) Example:

```
$ cat 1.png 2.png 3.png | wc -c
2053

$ if [ `cat 1.png 2.png 3.png | wc -c` -ge 2000 ] ; then echo yes greater than 2000 ; else echo not big enough
; fi
yes greater than 2000
```

Your answer here.

(c) (5 points)

We continue the scenario from the previous subquestion. Support or refute the general claim that Delta Debugging will work correctly for that notion of Interesting. Include a quote from the slides or readings (your choice, but indicate which one) to support your argument. (Beyond copying the quote, use at most four sentences.)

Your answer here.

Question 4. Design and Maintainability (10 points)

Answer the following questions about design and maintainability.

(a) (5 points)

Consider Liu et al.'s "Automatic Generation of Pull Request Descriptions" https://eecs481.org/readings/automaticpr.pdf. In class, we discussed **What** and **Why** as categories of documentation. Support or refute the claim that Liu et la.'s special handling of out-of-vocabulary (OOV) words will allow their approach to generate higher-quality Why information for pull request summaries. Give an example from the slides of the sort of Why documentation that you think would be handled well (or not) by Liu et al.'s method and indicate which slide number it corresponds to in https://eecs481.org/lectures/se-17-designmaint.pdf. Include a quote from the Liu et al. paper to support your argument. (Beyond copying the quote, use at most four sentences.)

LATE

minutes remaining

Hide Time

Manual Save

Navigation

- Question 1
- Question 2
- Question 3
- Question 4
- Question 5
- Question 6
- Question 7
- Question 8
- Extra Credit
- Pledge & Submit

Your answer here.

(b) (5 points)

Consider the driver, mutate and subject Python files from the Homework #3 exercise https://eecs481.org/hw3/hw3.zip. Consider the **Observer** (or publish-subscribe) software design pattern. Support or refute the claim that the driver should use the Observer design pattern to improve maintainability when running each generated mutant against the test suite. A subscriber could receive a push-style update() for each test outcome and note when exactly one test has failed for a mutant (and thus that the mutant is neither too strong nor too weak in that Homework's definition). Include a quote from the slides or course readings (your choice, but indicate which one) to support your argument. (Beyond copying the quote, use at most four sentences.)

Your answer here.

Question 5. Requirements and Elicitation (10 points)

Answer the following questions about requirements and elicitation.

(a) (5 points)

Consider the notion of a **verifiable non-functional requirement** from slide 38 of https://eecs481.org/lectures/se-13-req.pdf. Support or refute the claim that the Homework #2 specification at https://eecs481.org/hw2.html contains a verifiable non-functional requirement that the student (you) is instructed to follow. If you choose to support, quote the text showing the requirement and justify your answer. If you choose to refute, quote the text showing a requirement that comes close to qualifying but officially does not, and justify your answer. (Beyond quoting from the text, use at most four sentences.)

Your answer here.

(b) (5 points)

Consider the concept of a **Conflict** from requirements elicitation https://eecs481.org/lectures/se-14-valid.pdf. Consider the Homework #1B specification at https://eecs481.org/hw1.html and the Bjarnason et al. reading at https://eecs481.org/readings/bjarnason-re-vv.pdf. Support or refute the claim that RE conflict resolution on the word "error" would likely result in higher coverage for student test suites in HW1B. Include a quote from the reading and a quote from the specification to support your argument. (Beyond copying the quotes, use at most four sentences.)

Your answer here.

Question 6. Fault Localization (11 points)

Answer the following questions about fault localization.

(a) (5 points)

Consider Table I of the Pearson et al. reference from Homework #5: https://eecs481.org/hw5/fault_localization_effectiveness_icse_2017.pdf. The two bottom-most rows report on over 3,000 real and artificial faults in programs like JFreeChart (seen in Homeworks #1 and #2). They note that "Our results (bottom) uphold previous results on

LATE

minutes remaining

Hide Time

Manual Save

Navigation

- Question 1
- Question 2
- Question 3
- Question 4
- Question 5
- Question 6
- Question 7
- Question 8
- Extra Credit
- Pledge & Submit

artificial faults but not on real faults”.

What is an example of a previous result on artificial faults that they uphold? What formal software engineering concept (use its precise name) is generally called into question by the disparity between their results on artificial and real faults?

Include a quote from the relevant slides or course readings (your choice, but indicate which one) to support your argument. (Beyond copying the quote, use at most four sentences.)

Your answer here.

(b) (6 points)

Consider the Barinel Fault Localization algorithm from the bottom right of page 3 of https://eecs481.org/hw5/fault_localization_effectiveness_icse_2017.pdf. Its formula is:

```
Suspiciousness(s) = 1 - (passed(s) / (passed(s) + failed(s)))
```

(Reminder: passed(s) is the number of passed test cases that executed statement s.) Consider the following Python code with certain statements identified by calls to print():

```
def spider(x,y):
        print("1")
        if x < y:
                print("2")
        elif x > y:
                print("3")
        while x > 4:
                print("4")
                x = x / 2
                if (x == y):
                        print("5")
                        abort() # test fails if this is reached
        if x % 2 == y % 2:
                print("6")

```

With respect to the following six tests (i.e., test input values for x and y):

```
0,0
1,2
11,22
7,3
3,9
8,4
```

What is the Barinel Suspiciousness score for each identified statement? Format your answer as a Python list of numbers using at most 3 figures after any decimal point. Example: `[ 1.0, 0.667, -1.333, 0.0, 0.5, -1.0 ]`

If you submit a non-empty answer for this question, it must be entirely correct: **no partial credit** will be awarded. (If you do not submit an answer for this question, you receive the usual amount for leaving it blank.)

Your answer here.

Question 7. Software Engineering Cognition (15 points)

Answer the following questions about cognition and software engineering.

(a) (6 points)

Consider the individual performance results from Sackman et al.’s *Exploratory Experimental Studies Comparing Online and Offline Programming Performance*, and from Dougherty and Thadani’s *Economic Value of Rapid Response Time*. These are reproduced in the early slides of https://eecs481.org/lectures/se-19-productive.pdf. Consider Begel and Zimmermann’s report from Microsoft, Analyze This! 145 Questions for Data Scientists in Software Engineering (https://eecs481.org/readings/datase.pdf).

LATE

minutes remaining

Hide Time

Manual Save

Navigation

- Question 1
- Question 2
- Question 3
- Question 4
- Question 5
- Question 6
- Question 7
- Question 8
- Extra Credit
- Pledge & Submit

Identify two areas related to individual productivity: one in which the Begel and Zimmermann report is in alignment with one of the two productivity readings and one in which it opposes them. For each, include a different quote from the slides or course readings (your choice, but indicate which one) to support your argument. (Beyond copying the quotes, use at most six sentences.)

Your answer here.

(b) (5 points)

Consider McLoughlin et al.'s *Programmers' Visual Attention on Function Call Graphs During Code Summarization*, as described starting on slide 22 of https://eecs481.org/lectures/se-num-brain.pdf (and optionally available in full at https://yuhuang-lab.github.io/paper/huang-ase25.pdf). Then consider Graham et al.'s *gprof: a Call Graph Execution Profiler* https://eecs481.org/readings/gprof.pdf, which describes **flat** profiles in Section 5.1.

Support or refute the claim that flat profiles should be used to support developers during summarization. Include, as evidence, a quote from the slides or course readings (your choice, but indicate which one) to support your argument. (Beyond copying the quote, use at most four sentences.)

Your answer here.

(c) (4 points)

You are working at a company that has collected COCOMO-style project time cost data. For this question, use the COCOMO numbers presented on Slide 34 of https://eecs481.org/lectures/se-02-risk.pdf.

You are managing a team that is being asked to make a "Very High" complexity project with a "Nominal" required turnaround time. In addition:

As part of your requirements elicitation process, you have conveyed prototypes to the customer that highlight tradeoffs in quality properties. The customer has reported that either of the following two configurations would be acceptable:

- A. The delivered software must comply with "Extra high" run-time performance constraints and with "Nominal" memory constraints
- B. The delivered software must comply with "High" run-time performance constraints and with "Extra high" memory constraints

Similarly, you have two options for how you will hire and train your team of developers (cf. slide 50 of https://eecs481.org/lectures/se-num-brain.pdf):

- C. You can hire developers with "Low" software engineer capability.
- D. You can hire developers with "Low" software engineer capability, but subject them to a language ability training course. This training course takes time and slows down project delivery by 1.25x, but has the overall effect of giving your team "High" software engineer capability.

Indicate which software process decisions you should make (A or B, C or D) to minimize the predicted time cost of the project. Format your answer as a Python list containing two strings and a number rounded to two digits after the decimal, where the rounded number represents the total COCOMO-predicted time. Example: `["A", "C", 9.12]` would mean "choose option A, choose option C, and the final time cost will be 9.12".

Your answer here.

Question 8. AI and Code Synthesis (15 points)

Answer the following questions about AI and code synthesis.

(a) (5 points)

A hypothetical new French software engineer wants to use generative AI to write a procedure. She would like a procedure that drops the fourth element of a list. In French, this might be phrased as "déposer le quatrième".

LATE

minutes remaining

Hide Time

Manual Save

Navigation

- Question 1
- Question 2
- Question 3
- Question 4
- Question 5
- Question 6
- Question 7
- Question 8
- Extra Credit
- Pledge & Submit

Here is an actual interaction with ChatGPT 5.0:

- Prompt: Can you write me a small Python procedure deposerquat() that accepts a list X of numbers and returns a list Y. Here are some example tests it must comply with: deposerquat([1,2,3,3]) = [1,2,3] and deposerquat([8,6,7,7]) = [8,6,7] .

  ChatGPT: It looks like `deposerquat()` should return the input list with any trailing run of duplicates collapsed to a single value.

  ```
  [1,2,3,3] → [1,2,3] (last two 3’s become one 3)
  [8,6,7,7] → [8,6,7] (last two 7’s become one 7)
  ```

  Here is a simple Python procedure that does exactly that:

  ```
  def deposerquat(X):
      if not X:
          return []
      Y = X[:]  # copy
      # Remove duplicates at the *end* until the last two differ
      while len(Y) > 1 and Y[-1] == Y[-2]:
          Y.pop()
      return Y
  ```

Unfortunately, the result does not match her goals, even though she used formal tests to communicate her intention.

1. One friend tells her that she should provide more and better tests, such as by using the test input generation approaches in Homework #2 (https://eecs481.org/hw2.html).
2. Another friend tells her that she should use code inspection and carefully read the provided code (as in https://eecs481.org/lectures/se-07-codereview.pdf).
3. A third friend suggests that she should focus on requirements elicitation techniques to avoid terminology conflicts (as in https://eecs481.org/lectures/se-14-valid.pdf).
4. A fourth friend suggests that she use symbolic program synthesis techniques that are not based on large language models (as in https://eecs481.org/lectures/se-24-repair.pdf or https://eecs481.org/lectures/se-26-ai-1.pdf).

For each one of these four proposed approaches, give a reason why it is unlikely to work (i.e., could reasonably fail) for a new software developer. Use at most eight sentences.

Your answer here.

LATE

minutes remaining

Hide Time

Manual Save

Navigation

- Question 1
- Question 2
- Question 3
- Question 4
- Question 5
- Question 6
- Question 7
- Question 8
- Extra Credit
- Pledge & Submit

(b) (5 points)

Consider Endres et al.'s *Can Large Language Models Transform Natural Language Intent into Formal Method Postconditions?* (https://people.cs.umass.edu/~mendres/papers/CanPostconditions.pdf). Research Question 1 in that Microsoft report concludes the following about their `nl2postcond` technique:

- LLMs are good at producing correct postconditions from informal natural language specifications. All prompt variants generate a correct postcondition for at least 77% and up to 96% of problems. …

In HW2, you used Randoop to generate postconditions automatically (https://eecs481.org/hw2.html).

On the surface, these may seem like very similar tools. They both reduce software maintenance costs by making postconditions automatically. In practice, what they require of you may be more important than slight differences in their numerical accuracies — especially since those accuracies may vary from year to year as new models and improvements are developed. Address each of the following 8 questions:

1. Does Randoop need to see your source code to operate or to obtain good results? (yes/no)
2. Does nl2postcond need to see your source code to operate or to obtain good results? (yes/no)
3. Does Randoop normally run your code? (yes/no)
4. Does nl2postcond normally run your code? (yes/no)
5. Does Randoop need prose text from you describing desired behavior? (yes/no)
6. Does nl2postcond need prose text from you describing desired behavior? (yes/no)
7. How does Randoop determine if a generated test is positive or negative? (1-2 sentences)
8. How does nl2postcond determine if a postcondition is correct or not? (1-2 sentences)

(Answer all eight in order in the same text box.)

Your answer here.

Extra Credit

(1) Name a course staff member (Derek, Hanchi, Jesse, Priscila, Rohit, Sathvika, or Wes) and list something that person did that helped you learn. (1 point)

Your answer here.

(2) What is your least favorite part of the class so far? Alternatively, what improvement would you suggest for future semesters? (1 point)

Your answer here.

(3) Choose any optional reading (it must say `Optional:` before it on the Lectures page) or any post tagged `long-instructor-post` on Piazza that was not about course logistics (i.e., "Willpower ..." or "Persuading those in Power" or "Hiring, leetcode ..." would count, but "Exam 1, ChatGPT ..." or "Course Expectations ..." would not). Identify it and demonstrate to us that you have read it critically by relating it to your own personal experience, going *beyond* a Generative AI Summary. (We may not give credit if your answer looks like an impersonal AI summary.) (2 points)

Your answer here.

(4) Choose any other optional reading (it must say `Optional:` before it on the Lectures page) or any post tagged `long-instructor-post` on Piazza that was not about course logistics (i.e., "Willpower ..." or "Persuading those in Power" or "Hiring, leetcode ..." would count, but "Exam 1, ChatGPT ..." or "Course Expectations ..." would not). Identify it and demonstrate to us that you have read it critically by relating it to your own personal experience, going *beyond* a Generative AI Summary. (We may not give credit if your answer looks like an impersonal AI summary.) (2 points)

Your answer here.

(5) Which Generative AI tool(s) and version(s) did you use on this exam, if any? For which parts of the exam did you find them

(or speculate that they would be) helpful or unhelpful? (Remember, free GenAI is allowed, so this is not cheating. This is to help us improve the course, not to get you in trouble.) If you used any other sources that you are uncertain about, you can cite them here. (1 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.

LATE

minutes remaining

Hide Time

Manual Save

Navigation

- Question 1
- Question 2
- Question 3
- Question 4
- Question 5
- Question 6
- Question 7
- Question 8
- Extra Credit
- Pledge & Submit