

Python Patching Exit Survey

Approaching The Problem

Q1: How did you usually choose the order to approach failed code objects?

1. Start from the top of the file and work down.
2. Start from the bottom of the file and work up.
3. Start from the smallest code object and work towards larger ones.
4. Start from the largest code object and work towards smaller ones.
5. Start from the easiest code objects to fix and work towards harder ones.
6. Start from the hardest code objects to fix and work towards easier ones.
7. Start from main and follow function calls.
8. Other. (Please describe your approach)

Q2: How did you usually approach fixing issues within one failed code object?

1. Start from the top of the code object and work down.
2. Start from the bottom of the code object and work up.
3. Start from the easiest issues to fix and work towards harder ones.
4. Start from the hardest issues to fix and work towards easier ones.
5. Start with statement-level bytecode errors and handle control flow later.
6. Start with control flow and handle statement-level bytecode errors later.
7. Start from the top of the function and work in control-flow order (i.e., following jumps).
8. Other. (Please describe your approach.)

Q3: Rate your agreement with the following statements

a. My approach to patching significantly changed as I completed more challenges.

Strongly Disagree	Disagree	Somewhat Disagree	Neither Agree nor Disagree	Somewhat Agree	Agree	Strongly Agree
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

b. My approach to patching was influenced by a discussion with other students.

Strongly Disagree	Disagree	Somewhat Disagree	Neither Agree nor Disagree	Somewhat Agree	Agree	Strongly Agree
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Q4: Briefly describe how your approach to patching evolved as you completed challenges.

Q5: How often did you use tools outside of PyLingual to complete challenges?

Never	Rarely	Occasionally	Sometimes	Often	Very Often	Always
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Q6: If you used any external tools, list the tools you used.

Error Identification

Q7: How often did you use each of the following strategies to locate the source code section responsible for a particular error?

a. Matching line numbers between bytecode and source code.

Never	Rarely	Occasionally	Sometimes	Often	Very Often	Always
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

b. Recognizing patterns in the bytecode instructions.

Never	Rarely	Occasionally	Sometimes	Often	Very Often	Always
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

c. Searching for specific opcodes or constants related to the error (e.g., CALL_FUNCTION, JUMP_ABSOLUTE).

Never	Rarely	Occasionally	Sometimes	Often	Very Often	Always
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

d. Comparing expected vs. actual decompilation results.

Never	Rarely	Occasionally	Sometimes	Often	Very Often	Always
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Q8: If you used other strategies, please briefly describe them.

Q9: Rate your agreement with the following statement: The relationship between the bytecode and decompiled source was difficult to understand.

Strongly Disagree	Disagree	Somewhat Disagree	Neither Agree nor Disagree	Somewhat Agree	Agree	Strongly Agree
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Q10: How often were there cases where a single-bytecode instruction change led to a large difference in the decompiled source?

Never	Rarely	Occasionally	Sometimes	Often	Very Often	Always
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Q11: When faced with an error, did you tend to reason about the bytecode structurally (as a sequence of instructions) or functionally (as higher-level Python logic)?

1. Mostly structurally.
2. Mostly functionally.
3. A mix of both, depending on the situation.

Difficulty Estimation

Q12: How much time did you spend on average to complete a challenge?

1. less than 30 min
2. 30 min ~ 1 hour
3. 1 hour ~ 2 hours
4. 2 hours ~ 4 hours
5. 4 hours ~ (more)

Q13: What was the longest amount of time you took to complete a single challenge?

1. less than 30 min
2. 30 min ~ 1 hour
3. 1 hour ~ 2 hours
4. 2 hours ~ 4 hours
5. 4 hours ~ (more)

Q14: What step of patching took the most time/effort?

1. Identifying syntax errors.
2. Identifying bytecode errors.
3. Locating bytecode errors in the source code.
4. Modifying the source code to fix a given bytecode error.
5. Identifying and using external tools (e.g., pydisasm, cfg.py, etc.)

Q15: How difficult was correcting following decompiler errors:

a. Incorrect control flow (e.g., misplaced loops, conditionals, boolean expressions).

Very Difficult	Difficult	Somewhat Difficult	Moderate	Somewhat Easy	Easy	Very Easy
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

b. Incorrect exception handling.

Very Difficult	Difficult	Somewhat Difficult	Moderate	Somewhat Easy	Easy	Very Easy
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

c. Variable misassignments or incorrect scope handling.

Very Difficult	Difficult	Somewhat Difficult	Moderate	Somewhat Easy	Easy	Very Easy
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

d. Missing or extra functions.

Very Difficult	Difficult	Somewhat Difficult	Moderate	Somewhat Easy	Easy	Very Easy
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

e. Missing or extra bytecode in functions.

Very Difficult	Difficult	Somewhat Difficult	Moderate	Somewhat Easy	Easy	Very Easy
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

f. Incomplete statements.

Very Difficult	Difficult	Somewhat Difficult	Moderate	Somewhat Easy	Easy	Very Easy
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Q16: Please list any other kinds of errors that were very difficult to correct, if any.

Q17: How often did you understand the error but still struggle to implement the fix in Python?

Never	Rarely	Occasionally	Sometimes	Often	Very Often	Always
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Q18: How often did you feel that you completely understood a function, but still needed to try several equivalent implementations to achieve perfect decompilation?

Never	Rarely	Occasionally	Sometimes	Often	Very Often	Always
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Overcoming Challenges

Q19: How often did you feel stuck on a challenge?

Never	Rarely	Occasionally	Sometimes	Often	Very Often	Always
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Q20: What was your main strategy when stuck?

1. Re-reading the bytecode.
2. Experimenting with different Python constructs.
3. Looking for patterns in other functions.
4. Guessing and checking with syntax corrections.
5. Asking LLMs (e.g., ChatGPT) for help.
6. Asking others for help.
7. Other. (Please describe)

Q21: Did any of the errors feel particularly frustrating or unfair? If so, why?

Quality of PYC Challenges and Operation

Q22: What background information would you provide to teach a novice reverser how to repair PYC decompilations?

Q23: Did you encounter any usability issues in the decompiler interface?

Q24: Do you have any suggestions for improving the PyLingual IDE?

Motivation and Understanding

Q25: What are the key differences between regular binary reversing (e.g., IA-32, AMD64) and PYC reversing?

Q26: Do you see any value in repairing PYC decompilations? Would you go out of your way to patch a PYC and why?

Q27: Rate your agreement with the following statement: The PYC patching assignment helped me better understand malware.

Strongly Disagree	Disagree	Somewhat Disagree	Neither Agree nor Disagree	Somewhat Agree	Agree	Strongly Agree
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>