CSE 8A/6R Fall 2025 Homework 3

Due: Thursday, October 23, 2025 before 11:59pm

Full Name: _____ PID: _____

Homework 3 topics covered:
1. For loops, 2. While loops

5 core competencies:
1. Code Tracing, 2. Code Comprehension, 3. Code Writing, 4. Debugging, and 5. Testing

Accessing printed worksheets: we typically have homework copies available during:

1. In-person office hours (see: https://cse8a.site/calendar)
2. Fridays 10am - 2pm in room B250 (where in-person study groups are happening)
3. Mondays 10am - 6pm in room B250 (where in-person study groups are happening)
4. Wednesdays 10am - 6pm in room B250 (where in-person labs are happening)
5. Tuesday lectures, in the lecture hall

Where B250 refers to the room in the CSE basement floor. We might have additional copies in B250 outside of the above windows, but this is not guaranteed.

Submission

Questions 1 - 5:

- **Must be Handwritten** in the given boxes only.
- Scanned/uploaded to Gradescope **Homework 3 Questions 1 - 5** assignment.
- *If any of the questions 1 - 5 are typed, then your work will be considered **ungradable**.*

Question 6 (process journal):

- The **process journal** can be either handwritten or **typed (preferred)** into a document.
- Uploaded to Gradescope **Homework 3 Question 6 Process Journal** assignment.

When you submit your work, **please make sure to assign the pages to the questions**. Otherwise, your work will be considered **ungradable**.

Point breakdown for Homework 3:

- **Exemplary (3 homework points):** All problems receive a "Pass."
- **Satisfactory (2 homework points):** At least 4 out of 6 questions receive a "Pass."
- **Needs Improvement (1 homework point):** At least 1 out of 6 questions receive a "Pass."
- **Incomplete (0 homework points):** No problems receive a "Pass," or the assignment was not submitted, or the assignment is ungradable (e.g., illegible handwriting), or if the submission does not follow the instructions (e.g., homework is typed instead of handwritten, not answering in the boxes provided, not assigning pages to the question).

1. [Code Comprehension] In the following scenarios, which loop will you use and why?

A) You want to keep asking the user for a password until they enter the correct one.

Answer (circle one): For loop | While loop

Explanation:

B) Check each letter in a word to see if it's a vowel.

Answer (circle one): For loop | While loop

Explanation:

C) In a horror game, allow the user to walk down the hallway forever unless the user turns around and walks in the opposite direction.

Answer (circle one): For loop | While loop

Explanation:

2. [Code Tracing] What will be the output (printed) of running the following code? When showing your work, show how all values assigned to variables, including the variable `num`, `i` or `song`, change.

A)

`sum = 0`
`for num in range(2, 8, 2):`
`  num_tripled = num * 3`
`  sum += num_tripled`
`print(sum)`
	Output:

Show your work:	

B)

`word = "hello"`
`vowels = "aeiou"`
`for i in range(len(word)):`
`  if word[i] in vowels:`
`  to_print = word[i] * i`
`  print(to_print)`
	Output:

Show your work:	

C)

```
song_names = ["1989", "Reputation", "the 1", "Fearless", "Eras"]
for song in song_names:
   name_length = len(song)
   if name_length == 4:
       print(song)
```

Show your work:

Output:

3. [Debugging] Oh no, infinite loops! For the following question, we wanted to print the numbers 5, 3 and 1, but we got into an infinite loop instead. Explain why this happens and how to fix it.

Note: if you encounter infinite loops while programming, you can terminate the program by pressing `control + C` on your keyboard.

Code:	Expected output:	Actual output:

`number = 5`
`while number != 0:`
`  print(f"number = {number}")`
`  number = number - 2`
	number = 5
	number = 3
	number = 1
		number = 5
		number = 3
		number = 1
		number = -1
		number = -3
		… (keeps running!)

A) Explain: why did we enter an infinite loop?

B) Rewrite the corrected version of the above code in the box below. Your corrected code must still use a **while loop** and produce the expected output.

4. Hailey is working in the study group on two questions:

Question A: print numbers 0, 1, 2

Question B: print numbers 0, 1, 2, 4, 5 (skipping 3)

Masha walked by and saw Hailey working on these two questions, and told Hailey that this question can be solved by changing just one line, using the knowledge they learned in class: `break` and `continue`.

NOTE: The following code is the same for both questions A and B. You have to find out what will you fill in for the placeholder "`# fill this line`" for the two different scenarios shown below.

```
for i in range(6):
    if i == 3:
        # fill this line
    print(i)
```

Help Hailey fill in the code above using the following answer options and explain why it works!

Question A) Goal is to print numbers 0, 1, 2

Answer (circle one): `break` | `continue`

Explanation:

Question B) Goal is to print numbers 0, 1, 2, 4, 5 (skipping 3)

Answer (circle one): `break` | `continue`

Explanation:

5. [Code Tracing, Code Comprehension] Consider the following code and an attempt at running the code

```
def guess_number():
    print("Please think of a number between 1 and 100 (both inclusive).")
    print("I will try to guess it!")
    # Set the lower and upper bounds
    low = 1
    high = 100

    while True:
        guess = (low + high) // 2
        print(f"Is your secret number {guess}?")

        feedback = input("h if guess is too high, l if too low, c if correct: ")

        if feedback == 'h':
            high = guess - 1  # Adjust the upper bound
        elif feedback == 'l':
            low = guess + 1   # Adjust the lower bound
        elif feedback == 'c':
            print(f"Great! I guessed your number: {guess}.")
            break

# Run the game
guess_number()
```

Output when the number in the user's mind is 12:

```
Please think of a number between 1 and 100 (inclusive).
I will try to guess it!
Is your secret number 50?
h if guess is too high, l if too low, c if correct: h
Is your secret number 25?
h if guess is too high, l if too low, c if correct: h
Is your secret number 12?
h if guess is too high, l if too low, c if correct: c
Great! I guessed your number: 12.
```

A) Show how the variables in the function `guess_number()` change when we run the code to produce the output above.

B) How does the `guess_number` procedure stop running?

C) Explain in plain English (you can also draw, if that is helpful): how does this function guess a number the user is thinking about?

6. [Code Writing, Testing] Calculating the total amount of points the player earned in a game!

For this question, you are submitting a *process journal*. For detailed instructions, please check the instruction document: https://tinyurl.com/ProgrammingProcessJournal. Be sure to read this before you start implementing!

You are designing a 2D platformer game where the player can collect a gold coin (2 points), a silver coin (1 point), or fall into a trap and lose one point (-1 point). Design a program to calculate the total number of points that the player collected!

Name of the function: `calculate_points`
Parameter:

- `events`: *a list of strings*, representing the events that the player encountered.
  - `"gold"`: the gold coin, which should earn 2 points
  - `"silver"`: the silver coin, which should earn 1 point
  - `"trap"`: a trap, which should *reduce* 1 point
  - Any other string: ignore, 0 point.

Data type of return value: integer (int), representing the total amount of points (the value could be negative or positive).

Example of calling and testing the function:

```
events_sample = ["gold", "silver", "trap", "mushroom", "trap"]
print(calculate_points(events_sample))  # Output should be 1
```

Explanation: points = 2 + 1 - 1 + 0 - 1 = 1