

Homework Assignment 2 (100 Points)
CSE 464, Fall 2025
SCAI, Arizona State University
Due: By Saturday Sep 27th 11:59 pm

Please see the submission instructions given on the last page

Note: This is individual homework, DO NOT collaborate, NO AI related tools are allowed

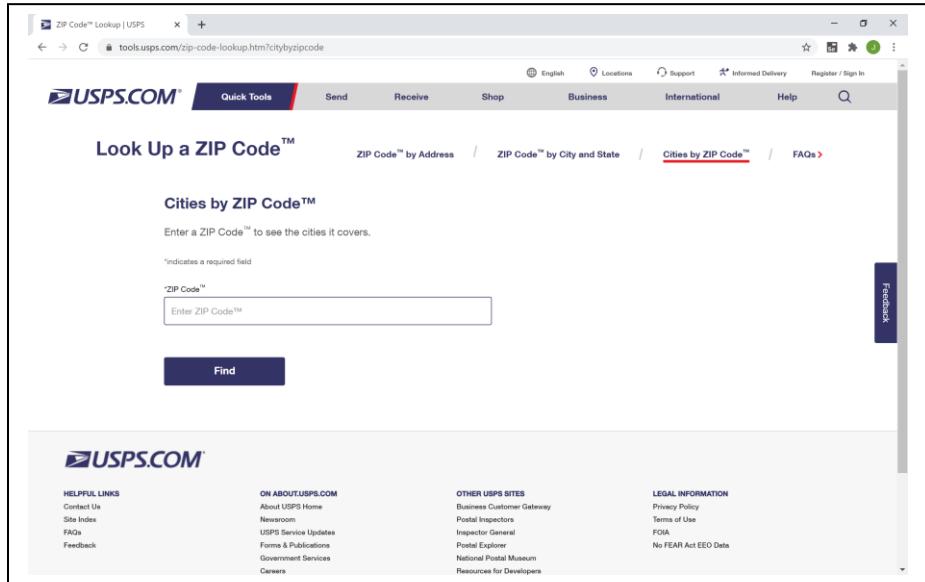
Introduction: This assignment helps you to reinforce the topics discussed in the class including

- a) Basic terminology of Testing and Quality Assurance, Software Quality, Software Quality Assurance, and Intro to Testing.
- b) Categories of testing and equivalent class testing.
- c) Writing Junit test cases for test automation

1. [25 Points] Following is the web interface of the zip code lookup functionality of USPS (<https://tools.usps.com/zip-code-lookup.htm?citybyzipcode>)

Input: ZIP code

Output: List of cities



The screenshot shows a web browser window for the USPS ZIP Code Lookup. The URL in the address bar is <https://tools.usps.com/zip-code-lookup.htm?citybyzipcode>. The page title is "ZIP Code™ Lookup | USPS". The top navigation bar includes links for Quick Tools, Send, Receive, Shop, Business, International, Help, and Register / Sign In. The main content area has a header "Look Up a ZIP Code™" with sub-links for ZIP Code™ by Address, ZIP Code™ by City and State, Cities by ZIP Code™ (which is underlined in red), and FAQs. Below this is a form titled "Cities by ZIP Code™" with a placeholder "Enter a ZIP Code™ to see the cities it covers." A note says "*indicates a required field". There is a text input field labeled "ZIP Code™" with "Enter ZIP Code™" inside, and a "Find" button. At the bottom of the page, there is a footer with links for "HELPFUL LINKS" (Contact Us, Site Index, FAQs, Feedback), "ON ABOUT.USPS.COM" (About USPS Home, Newsroom, USPS Service Updates, Forms & Publications, Government Services, Careers), "OTHER USPS SITES" (Business Customer Gateway, Postal Inspectors, Inspector General, Postal Explorer, National Postal Museum, Resources for Developers), and "LEGAL INFORMATION" (Privacy Policy, Terms of Use, FOIA, No FEAR Act EEO Data).

- a) Use the techniques and approaches discussed in the class to determine equivalent classes to test this functionality. Clearly state your approach and equivalent classes(partitions). Points will be given based on the effectiveness and quality of your test cases (part b) and equivalent partitions

- b) Develop test cases (a table with test case #, test value, and expected output as columns) based on your equivalent classes and test the system for correctness. The number of test cases should be between 10 and 15. If you have large number of test cases based on your approach, list max 15 test cases and briefly explain how you would obtain other test cases.
- c) Do you think USPS has a good implementation of this functionality?

2.[75 Points] BMR Calculation Program Description

Consider an application that accepts three inputs: **age (years)**, **weight (pounds)**, and **height (inches)** to calculate the Basal Metabolic Rate (BMR) of a person.

The program calculates the BMR value using the **Mifflin-St Jeor Equation**:

Men:

$$\text{BMR} = 66 + (6.23 \times \text{weight in pounds}) + (12.7 \times \text{height in inches}) - (6.8 \times \text{age in years})$$

Women:

$$\text{BMR} = 655 + (4.35 \times \text{weight in pounds}) + (4.7 \times \text{height in inches}) - (4.7 \times \text{age in years})$$

Program Behavior

Based on the values of the input variables or the calculated BMR, the application should provide relevant output:

1. Invalid Inputs:

- o Age, height, or weight cannot be negative: $\text{age} < 0$ or $\text{height} < 0$ or $\text{weight} < 0$
- o Age, height, or weight cannot be zero: $\text{age} = 0$ or $\text{height} = 0$ or $\text{weight} = 0$

2. BMR Categories:

- o **Low Metabolism:** $\text{BMR} < 1200 \text{ kcal/day}$
- o **Moderate Metabolism:** $1200 \leq \text{BMR} < 1800 \text{ kcal/day}$
- o **High Metabolism:** $1800 \leq \text{BMR} < 2500 \text{ kcal/day}$
- o **Very High Metabolism:** $\text{BMR} \geq 2500 \text{ kcal/day}$

Valid Input Ranges

- **Age:** $0 < \text{age} < 120$ years
- **Height:** $0 < \text{height} < 100$ inches
- **Weight:** $0 < \text{weight} < 1000$ pounds

The program should take gender as an input ("male" or "female") to apply the correct BMR formula.

Suppose you are going to test this application using EP and Boundary Value testing approaches

- a) Use the techniques and approaches discussed in the class to determine equivalent classes to test this functionality. Clearly state your approach and equivalent classes(partitions).
- b) Develop test cases (a table with test case #, test value, and expected output as columns) based on your equivalent classes and test the system for correctness. Use Weak Robust classification in developing test cases.
- c) Develop Junit Test Cases and Test the Assignment2.java program using part b above. How much code coverage do you get from your test cases (Give the screenshot of the code coverage)

Note: Points will be given based on the effectiveness and quality of your test cases (part b) and equivalent partitions

- d) Identify boundary values for the three inputs age, height, and weight.
- e) Develop test cases (a table with test case #, test value, and expected output as columns) for boundary values using Robustness approach.
- f) Develop Junit Test Cases and Test the Assignment2.java program using test cases in part e above. How much code coverage do you get from your test cases (Give the screenshot of the code coverage)

Submission Instructions

Use the Homework #2 Template to submit your solutions to HW2

NO LATE SUBMISSIONS WILL BE ACCEPTED