**Programming quizzes**

**Quiz 1**

**Write a program that takes as input the radius of a large circle and the radius of a small circle. Calculate the difference between the areas of these two circles.**

**(Let**

**= 3.14, r be the radius, then area =**

**r2)**

**Example:**

**Radius of the large circle: 10**

**Radius of the small circle: 5**

**Difference between their areas: 235.5**

*def calculate\_area\_difference(radius\_large, radius\_small):*

*# Define the value of pi*

*pi = 3.14*

*# Calculate the area of the large circle*

*area\_large = pi \* (radius\_large \*\* 2)*

*# Calculate the area of the small circle*

*area\_small = pi \* (radius\_small \*\* 2)*

*# Calculate the difference in area*

*difference = area\_large - area\_small*

*return difference*

*# Input from user*

*radius\_large = float(input("Radius of the large circle: "))*

*radius\_small = float(input("Radius of the small circle: "))*

*# Calculate and print the difference*

*difference = calculate\_area\_difference(radius\_large, radius\_small)*

*print("Difference between their areas:", difference)*

**A store pays its entry-level employees weekly salary based on their ages and length of contract if the age is 20:**

**Age Weekly Salary**

**14-15 343.44**

**16-17 457.92**

**18-19 610.56**

**20, 6 months or less contract 686.88**

**20, longer than 6 months contract 763.20**

**>20 850.34**

**Write a program that reads the age and length of the contract for an employee, and then prints the equivalent annual salary of the employee. (Assume 52 weeks per year.)**

**Example:**

**Age: 20**

**Contract length: 8 months**

**Annual salary: 39686.4**

*def calculate\_annual\_salary(age, contract\_length\_months):*

*# Define weekly salaries based on age and contract length*

*if 14 <= age <= 15:*

*weekly\_salary = 343.44*

*elif 16 <= age <= 17:*

*weekly\_salary = 457.92*

*elif 18 <= age <= 19:*

*weekly\_salary = 610.56*

*elif age == 20:*

*if contract\_length\_months <= 6:*

*weekly\_salary = 686.88*

*else:*

*weekly\_salary = 763.20*

*else: # age > 20*

*weekly\_salary = 850.34*

*# Calculate annual salary*

*annual\_salary = weekly\_salary \* 52*

*return annual\_salary*

*# Input from the user*

*age = int(input("Age: "))*

*contract\_length = int(input("Contract length (in months): "))*

*# Calculate and print the annual salary*

*annual\_salary = calculate\_annual\_salary(age, contract\_length)*

*print("Annual salary:", annual\_salary)*

**Input three integers a, b, and c. Sort them in descending order. With the sorted integers, print True if the differences between every two neighbouring numbers are the same. Otherwise, print False. (Python sort(), sorted(), max(), and min() functions are not allowed.)**

**Example:**

**a: 10**

**b: 30**

**c: 20**

**Output: True**

*def sort\_descending\_without\_sort(a, b, c):*

*# Initialize placeholders*

*highest = mid = lowest = None*

*# Find the highest number*

*if a >= b and a >= c:*

*highest = a*

*if b >= c:*

*mid, lowest = b, c*

*else:*

*mid, lowest = c, b*

*elif b >= a and b >= c:*

*highest = b*

*if a >= c:*

*mid, lowest = a, c*

*else:*

*mid, lowest = c, a*

*else:*

*highest = c*

*if a >= b:*

*mid, lowest = a, b*

*else:*

*mid, lowest = b, a*

*return highest, mid, lowest*

*def check\_equal\_differences(a, b, c):*

*# Sort the numbers in descending order without sort()*

*desc\_a, desc\_b, desc\_c = sort\_descending\_without\_sort(a, b, c)*

*# Check if the differences are the same*

*return (desc\_a - desc\_b) == (desc\_b - desc\_c)*

*# Input from the user*

*a = int(input("a: "))*

*b = int(input("b: "))*

*c = int(input("c: "))*

*# Print the result*

*print(check\_equal\_differences(a, b, c))*

**Quiz 2**

**Write a program that reads student IDs typed by the user until an ID less than 4 digits is entered, then prints the total number of IDs that were entered.**

**Example:**

**Enter an ID: 5154772**

**Enter an ID: 10438**

**Enter an ID: 2289937**

**Enter an ID: 389**

**4 IDs were entered.**

*def main():*

*ids\_entered = 0*

*while True:*

*try:*

*user\_input = input("Enter an ID: ")*

*student\_id = int(user\_input) # Convert input to integer*

*# Check if the ID has less than 4 digits*

*if student\_id < 10000:*

*break # Exit loop if ID is less than 4 digits*

*ids\_entered += 1 # Increment count of entered IDs*

*except ValueError:*

*print("Invalid input. Please enter a valid student ID (numeric).")*

*print(f"{ids\_entered} IDs were entered.")*

*if \_\_name\_\_ == "\_\_main\_\_":*

*main()*

**Write a function called mycase(chars, option) to return a copy of string chars with the case converted. The input argument option is a keyword (or keyworded) argument with a default value "lower". When option = "lower", covert all letters to lowercase; when option = "upper", covert all letters to uppercase; when option = "capitalize", convert the first character to uppercase and the rest lowercase. (Python built-in functions are allowed. Define the function only without calling it. Assume all characters in chars are alphabetic and there is at least one character.)**

*def mycase(chars, option="lower"):*

*if option == "lower":*

*return chars.lower()*

*elif option == "upper":*

*return chars.upper()*

*elif option == "capitalize":*

*return chars.capitalize()*

*else:*

*raise ValueError("Invalid option. Please use 'lower', 'upper', or 'capitalize'.")*

**Write a function that accepts two arguments, an integer, k, and a list of integers lst, and returns True if and only if there exists at least one pair of numbers selected from the list that differ by k, otherwise, return False. For example, given a list lst=[0, 1, 2, 4, 100], if k is 3 then return True, since 4 - 1 = 3; if k is 50, then return False.**

**Assume the function arguments are valid. Do not handle erroneous arguments. You are asked to write a function, not a complete program.**

*def find\_pair\_with\_difference\_k(k, lst):*

*# Create a set to store seen numbers*

*seen\_numbers = set()*

*for num in lst:*

*# Check if the current number's complement (num - k or num + k) is in the seen\_numbers set*

*if (num - k) in seen\_numbers or (num + k) in seen\_numbers:*

*return True*

*# Add the current number to the set*

*seen\_numbers.add(num)*

*# If no such pair is found after iterating through the list, return False*

*return False*

**Write a program that prompts for the names of a source file to read and a target file to write, and copy the content of the source file to the target file, but with all lines containing the colon symbol ‘:’ removed. Finally, close the files. You may test your code using the file in this link: test\_file.txtLinks to an external site. (the file can also be accessed in course Teams -> General -> Files -> Class Materials -> test\_file.txt)**

*def copy\_file\_with\_filter(source\_file\_name, target\_file\_name):*

*# Open source file for reading*

*try:*

*with open(source\_file\_name, 'r') as source\_file:*

*# Open target file for writing*

*with open(target\_file\_name, 'w') as target\_file:*

*# Read each line from source file*

*for line in source\_file:*

*# Check if line contains ':', if not, write to target file*

*if ':' not in line:*

*target\_file.write(line)*

*print(f"Content copied from '{source\_file\_name}' to '{target\_file\_name}' with lines containing ':' removed.")*

*except FileNotFoundError:*

*print(f"Error: One or both files '{source\_file\_name}' or '{target\_file\_name}' not found.")*

*# Prompt user for file names*

*source\_file\_name = input("Enter the source file name: ")*

*target\_file\_name = input("Enter the target file name: ")*

*# Call function to copy file content with filter*

*copy\_file\_with\_filter(source\_file\_name, target\_file\_name)*