|  |  |
| --- | --- |
| *Name: Christofer Piedra* | ***Total Points: 20*** |
| *Z number: Z23365973* |  |

COP 4045-001: Python Programming

COP 4045-002: Python Programming

FLORIDA ATLANTIC UNIVERSITY

Worksheet 1

NOTE:

* Each question carries 1 point.
* All solutions must be answered in the space labeled “solution”.
* Submit the completed worksheet in the form of a PDF.
* The file name must be *YourName\_Worksheet1.pdf*

**1. Think Before You Program (Designing a Program)**

You are tasked with creating a program that calculates the area of a circle. The program should:

* Take the radius as input from the user.
* Use the formula area = math.pi \* radius \*\* 2 to calculate the area.
* Display the result formatted to two decimal places.

**Question**: What are the steps to design the program for this task? Write the algorithm in plain English,

considering Rule 1: Think before you program. Then, implement the solution in Python.

**Hint**: Use the math module for π (math.pi) and consider input type conversion.

***Solution:***

*import math*

*radius = int(input("\nEnter the radius of a circle\n> "))*

*area = math.pi \* radius \*\* 2*

*print (f"\n Area is {area:.2f}\n")*

***Explanation:***

*1. Get the radius as input from user, assign it to value “radius”*

*2. Calculate area using area = math.pi \* radius \*\* 2*

*3. Use a print statement to display the area calculation (:.2f)*

**2. Error Handling and Debugging**

You are given the following code snippet:

x = 5

y = "10"

result = x + y

print(result)

**Question**: What error will occur when you run the code? Why does it occur, and how can you fix it to perform the addition of x and y?

***Solution:***

*There will be a value type error due to unsupported operant types. Fix it by removing the quotation marks from line 2*

***Explanation:***

*This error will occur because y is saved as a string, while x is saved as an integer. These value types can no both be used by the + at the same time*

**3. Using Functions**

Write a function reverse string(s) that takes a string as input and returns the string reversed.

**Question**: What will the following call print?

print(reverse\_string("Python"))

**Hint**: You can use string slicing or the reversed () function in Python.

***Solution:***

*def reverse\_string(word):*

*reversed\_word = ""*

*for letter in range(len(word) - 1, -1, -1):*

*reversed\_word += word[letter]*

*return reversed\_word*

*print(reverse\_string("Python"))*

***This print statement would output:***

***“nohtyP”***

**4. Modules and Libraries**

You are working on a project that requires you to calculate the square root of a number. Instead of writing the function yourself, you decide to use the math module.

**Question**: Write a Python script that:

* Imports the math module.
* Takes a positive number as input from the user.
* Calculates and prints the square root of that number.

**Hint**: Use math.sqrt().

***Solution:***

*import math*

*def get\_input():*

*while True:*

*try:*

*number = float(input("Enter a positive value\n> "))*

*if number > 0:*

*return number # Return only if it's a valid positive number*

*else:*

*print("\nYou inputted a non-positive number. Try again.")*

*except ValueError:*

*print("\nInvalid input. Please enter a number.")*

*def calculate(number):*

*sqrt = math.sqrt(number)*

*print(f"The square root of {number} is {sqrt}")*

*print("\n")*

*number = get\_input()*

*calculate(number)*

**5. Data Types and Conversion**

Consider the following code:

user\_input = input("Enter a number: ")

num = int(user\_input)

result = num \* 2

print(result)

**Question**: If the user enters "3.5" when prompted, what will happen? Modify the code to handle both integer and floating-point numbers correctly.

***Solution:***

*There is a error message stating:*

*ValueError: invalid literal for int() with base 10: '3.5'*

*I modified this code by:*

*1. Removing line 3*

*2. Type-casting user\_input to a float in line 1*

*3. 3. Replacing num with user\_input in line 3*

**6. Conditional Statements**

Write a Python program that:

* Asks the user for an integer input.
* If the number is even, print "Even".
* If the number is odd, print "Odd".

**Question**: What is the output of the program if the user enters 7?

***Solution:***

*while True:*

*try:*

*integer = input ("\nEnter an integer\n> ")*

*if integer.isdigit():*

*if float(integer) % 2 == 0:*

*print("Even")*

*else:*

*print("Odd")*

*else:*

*print("\nInvalid input. Please enter a positive integer.")*

*except ValueError:*

*print("\nInvalid input. Please enter a positive integer.")*

***Output if the user entered 7 is “Odd”***

**7. Loops and Iteration**

Write a Python program that:

* Takes a string as input.
* Prints each character of the string on a new line using a for loop.

**Question**: What will be the output if the user enters the string "Python"?

***Solution:***

*string = input("\n Enter a string of characters\n> ")*

*for i in range(len(string)):*

*character = string[i]*

*print(f"{character}")*

***If a user inputs “Python”, the output will be:***

***“P***

***y***

***t***

***h***

***o***

***n”***

**8. List Manipulation**

Consider the following list of numbers:

numbers = [10, 20, 30, 40, 50]

**Question**: Write a Python program that:

* Prints the second and fourth elements from the list.
* Adds 60 to the list.
* Removes the first element and prints the modified list.

***Solution:***

*numbers = [10, 20, 30, 40, 50]*

*print (f"\nSecond element of list called numbers is: {numbers[1]}")*

*print (f"Fourth element of list called numbers is: {numbers[3]}\n")*

*numbers.append(60)*

*numbers.pop(0)*

*print (f"{numbers}")*

**9. String Slicing and Manipulation**

Given the following string:

s = "Hello, World!"

**Question**: What is the output of the following code?

print(s[7:12])

print(s[:5])

print(s[-6:])

**Hint**: Consider string slicing syntax: s[start:end].

***Solution:***

**10. List Comprehension and Conditionals**

Write a Python program using list comprehension that:

* Takes a list of numbers as input.
* Creates a new list that contains only the even numbers from the original list.

**Question**: What will be the output of the following code?

numbers = [1, 2, 3, 4, 5, 6]

even\_numbers = [num for num in numbers if num % 2 == 0]

print(even\_numbers)

***Solution:***

*# Take input from the user and convert it into a list of integers*

*numbers = [list(map(int, input("Enter numbers separated by spaces: ").split()))]*

*# List comprehension to filter even numbers*

*even\_numbers = [num for num in numbers if num % 2 == 0]*

*# Print the result*

*print("Even numbers:", even\_numbers)*

***print(s[7:12]) - Outputs the word “World”***

***print(s[5]) - Outputs the comma “,”***

***print(s[-6]) - Outputs the letter “W”***

**11.** **Question 1: Using If-Else Statements**

Write a Python program that takes a user's age as input and prints the following:

* If the age is 18 or older, print "Adult".
* If the age is between 13 and 17, print "Teenager".
* If the age is between 3 and 12, print "Child".
* If the age is less than 3, print "Baby".

**Hint:** Use if, elif, and else statements to handle the different cases.

***Solution:***

# Take the user's age as an input, then output "Adult," "Teenager," "Child,", or "Baby" depending on their age

while True:

try:

age = float(input("\nEnter your age\n> "))

if age >=0:

if age >= 18:

print("Adult")

elif age >= 13 and age <= 17:

print("Teenager")

elif age >= 3 and age <= 12:

print("Child")

else:

print("Baby")

else:

print("\nInvalid Input, please enter a positive number")

except ValueError:

print("\nInvalid Input, please enter a positive number")

**12: Basketball Game Lead Calculation**

Implement the **Safe Lead** algorithm described in the prompt. Write a Python program that calculates if the lead in a basketball game is insurmountable.

* Take input for the number of points the leading team is ahead.
* Subtract 3 from the score.
* Add 0.5 if the leading team has the ball, subtract 0.5 if the trailing team has the ball.
* Square the result.
* If the result is greater than the number of seconds left in the game, print "Lead is Safe". Otherwise, print "Lead is Not Safe".

***Solution:***

*# Ensure correct value type*

*while True:*

*try:*

*# 1. Take Input for the number of points the team is ahead*

*lead = int(input("\nHow many points is the team ahead by\n> "))*

*if lead >= 0:*

*break*

*else:*

*print("\nInvalid input. Please enter a positive integer.")*

*except ValueError:*

*print("\nInvalid input. Please enter a positive integer.")*

*# 2. Subtract 3 from the score*

*lead -= 3*

*# Ensure correct value type*

*while True:*

*try:*

*# Determine who has the ball*

*possession = input("\nDoes the leading team have the ball? Enter Y or N\n> ").upper()*

*if possession == "Y" or possession == "N":*

*break*

*else:*

*print("\nInvalid input. Please enter 'Y' or 'N'.")*

*except Exception as e:*

*print(f"\nAn unexpected error occurred: {e}")*

*# 3. Add 0.5 if the leading team has the ball, subtract 0.5 if the trailing team has the ball*

*if possession == 'Y':*

*lead += 0.5*

*else:*

*lead -=0.5*

*# 4. Square the result*

*lead\_squared = lead \* lead*

*while True:*

*try:*

*# Determine how many seconds are left in the game*

*seconds\_left = int(input("\nHow many seconds are left in the game?\n> "))*

*if seconds\_left >= 0:*

*break*

*else:*

*print("\nInvalid input. Please enter a positive integer.")*

*except ValueError:*

*print("\nInvalid input. Please enter a positive integer.")*

*# 5. If the result is greater than the number of seconds left in the game, print "Lead is Safe". Otherwise, print "Lead is Not Safe"*

*if lead\_squared > seconds\_left:*

*print("\nLead is Safe\n")*

*else:*

*print("\nLead is Not Safe\n")*

**13: Factorial Calculation Using a While Loop**

Write a Python program that calculates the factorial of a number using a while loop. The program should:

* Prompt the user for a non-negative integer.
* Calculate the factorial of the number (e.g., 5! = 5 \* 4 \* 3 \* 2 \* 1).
* Display the result.

**Hint:** The factorial of n is calculated as n \* (n-1) \* ... \* 1. Be sure to use a while loop to calculate the product.

***Solution:***

*# Ensure correct value type*

*while True:*

*try:*

*# 1. Promput user for a non-negative number*

*number = float(input("\nEnter a non-negative number\n> "))*

*if number >= 0:*

*break*

*else:*

*print("\nInvalid input. Please enter a non-negative number.")*

*except ValueError:*

*print("\nInvalid input. Please enter a non-negative number.")*

*factorial = 1*

*# 2. Calculate the factorial of the number. Be sure to use a while loop to calculate the product*

*while number > 0:*

*factorial \*= number*

*number -= 1*

*# 3. Display the result*

*print(factorial)*

**14: Collatz Conjecture Sequence**

Write a Python program that takes an integer as input and applies the **Collatz Conjecture** algorithm to the number:

* If the number is even, divide it by 2.
* If the number is odd, multiply it by 3 and add 1.
* Continue applying this rule until the number becomes 1.
* Print the sequence of numbers generated.

**Hint:** Use a while loop to repeat the process until the number becomes 1.

***Solution:***

*while True:*

*try:*

*# Take an integer as input*

*integer = int(input("\nEnter an integer:\n> "))*

*if integer >= 1:*

*break*

*else:*

*print("\nInvalid input. Please enter a positive integer.")*

*except ValueError:*

*print("\nInvalid input. Please enter a positive integer.")*

*print ("")*

*while integer != 1:*

*if integer % 2 == 0:*

*integer/=2*

*else:*

*integer = integer \* 3 + 1*

*print (integer)*

*print ("")*

**15: Swap Two Numbers Without a Temporary Variable**

Write a Python program that takes two integers as input and swaps their values without using a temporary variable. You can use multiple assignment for this.

**Hint:** Use the syntax a, b = b, a to swap the values of a and b.

***Solution:***

print()

while True:

try:

# Take a non-zero integer as input

integer1 = int(input("Enter the first integer:\n> "))

integer2 = int(input("\nEnter the second integer:\n> "))

break

except ValueError:

print("\nInvalid input.")

# Swap values without using a temporary value

integer1, integer2 = integer2, integer1

print(f"\nAfter swapping:\nInteger 1 = {integer1}\nInteger 2 = {integer2}\n")

**16: Check for Perfect Number**

A **perfect number** is a number that is equal to the sum of its divisors, excluding itself. Write a Python program that:

* Prompts the user for a number.
* Determines if the number is a perfect number.
* Prints "Perfect Number" if the number is perfect, or "Not a Perfect Number" otherwise.

**Hint:** You can find divisors by iterating from 1 to n-1 and checking if each number divides n evenly.

***Solution:***

*# 2. Determines if the number is a perfect number*

*def perfect\_number(num):*

*return sum([i for i in range(1, num) if num % i == 0]) == num*

*while True:*

*try:*

*# 1. Prompts the user for a number*

*number = int(input("\nEnter a number:\n> "))*

*break*

*except ValueError:*

*print("\nInvalid input.")*

*# 3. Prints:*

*# A. "Perfect Number" if the number is perfect*

*if perfect\_number(number) == True:*

*print("Perfect Number\n")*

*# B. "Not a Perfect Number" if the number is not perfect*

*else:*

*print("Not a Perfect Number\n")*

**17: Nested If-Else for Discount Calculation**

Write a Python program that calculates the discount on a product based on the following conditions:

* If the product price is above $100, apply a 20% discount.
* If the price is between $50 and $100, apply a 10% discount.
* If the price is below $50, apply a 5% discount.
* If the price is exactly $0, print "Invalid price".

The program should take the price as input and print the final price after the discount.

***Solution:***

print()

while True:

try:

price = int(input("What is the price?\n> $"))

# 4. If price is exactly $0, print "Invalid Price"

if price == 0:

print("\nInvalid price.")

elif price < 0:

print("\nInvalid input.")

else:

break

except ValueError:

print("\nInvalid input.")

if price > 100:

discount = 0.2

elif price >= 50 and price <= 100:

discount = 0.1

# 3. If price is < $50, 5% discound

else:

discount = 0.05

print(f"\n\

Original Price: ${price}\n\

Discounted Price: ${price \* (1-discount)}\n\

")

**18: Prime Number Checker Using a While Loop**

Write a Python program that checks if a given number is prime. The program should:

* Take an integer input.
* Use a while loop to check if the number is divisible by any integer between 2 and the square root of the number.
* If the number is divisible by any of those integers, print "Not Prime"

***Solution:***

import math

# Input validation

while True:

try:

# 1. Take an integer as an input

integer = int(input("\nEnter an integer:\n> "))

if integer >= 1:

break

else:

print("\nInvalid input. Please enter a positive integer.")

except ValueError:

print("\nInvalid input. Please enter a positive integer.")

# Handle special cases

if integer == 1:

print("1 is neither prime nor composite.")

elif integer == 2:

print("2 is a prime number.")

else:

flag = False # Assume the number is prime

# Check divisibility from 2 to sqrt(integer)

for i in range(2, int(math.sqrt(integer)) + 1):

if integer % i == 0:

flag = True

break # No need to check further, number is not prime

if flag:

# 2. Print "Not Prime" if the number isn't prime

print("\nNumber is Not Prime\n")

else:

# 3. Print "Prime" if the number prime

print("\nNumber is Prime\n")

**19: Fibonacci Sequence Using a For Loop**

Write a Python program that prints the first n numbers of the Fibonacci sequence, where n is provided by the user. The program should:

* Use a for loop to generate the Fibonacci sequence.
* The first two Fibonacci numbers are 0 and 1, and each subsequent number is the sum of the previous two numbers.
* Print the sequence up to the n-th number.

***Solution:***

*print()*

*while True:*

*try:*

*# Prompt user*

*n = int(input("How many values into the Fibonacci Sequense should I print?\n> "))*

*if n > 0:*

*break*

*else:*

*print("\nInvalid input. Please enter a non-zero integer.")*

*except ValueError:*

*print("\nInvalid input. Please enter a non-zero integer.")*

*a = 0*

*b = 1*

*print(a)*

*# 1. Use a for loop to generate the Fibonacci sequence*

*for i in range(n-1):*

*# 2. Print the sequence up to the n-th number*

*print(b)*

*temp = a*

*a = b*

*b = temp + b*

**20: Using Boolean Expressions for Eligibility Check**

Write a Python program that checks if a person is eligible to vote based on the following conditions:

* The person must be at least 18 years old.
* The person must be a citizen of the country.

The program should prompt the user for their age and citizenship status (Yes/No). Based on these inputs, the program should print:

* "Eligible to vote" if the person is 18 or older and is a citizen.
* "Not eligible to vote" otherwise.

**Hint:** Use boolean expressions to combine the age and citizenship conditions.

***Solution:***

print()

while True:

try:

# Prompt user for age

age = int(input("Enter your age:\n> "))

if age > 0:

break

else:

print("\nInvalid input. Please enter a non-zero integer.")

except ValueError:

print("\nInvalid input. Please enter a non-zero integer.")

print()

while True:

try:

# 3. The program should prompt the user for their age and citizenship status (Yes/No). Based on these inputs

citizen = input("Are you a Citizen of this country? Enter 'Yes' or 'No'\n> ").upper()

if citizen == "YES" or citizen == "NO":

break

else:

print("\nInvalid input.")

except ValueError:

print("\nInvalid input.")

# 1. The person must be at least 18 years old

# 2. The person must be a citizen of the country

if age >= 18 and citizen == 'YES':

# A. "Elibible to vote" if the person is 18 or older and is a citizen

print("\nEligible to vote\n")

else:

# B. "Not Elibible to vote" otherwise

print("\nNot Eligible to Vote")