Assignment 10 Solution

Submitted by: Yameen Ali

Go to Github repository

Topics Covered

1. Conditional Statements
2. Error Handling
3. Function Definition
4. Discrete Fourier Transform (DFT)
5. Mathematical Operations
6. Complex Numbers
7. Poisson Distribution
8. Random Number Generation
9. Probability Mass Function
10. Data Structures (Lists, Tuples, Dictionaries)
11. Enumeration and Indexing
12. Zip Function
13. Random Selection
14. Randomization
15. Shuffling
16. Python Scripting
17. Code Commenting
18. Testing and Debugging

Question1:

You are developing a Python program for a student grading system. Implement a function calculate_grade that takes a student's score as input and returns their grade based on the following criteria:

▪ A score >= 90: Grade A ▪ 80 <= score < 90: Grade B ▪ 70 <= score < 80: Grade C ▪ 60 <= score < 70: Grade D ▪ score < 60: Grade F

However, if the input score is not a valid integer or is outside the range of 0 to 100, raise a ValueError with an appropriate error message (Invalid score. Score must be an integer between 0 and 100).

In [5]:
```
def calculate_grade(score):
    if not isinstance(score, int) or score < 0 or score > 100:
        raise ValueError("Invalid score. Score must be an integer between 0 and 100")

    if score >= 90:
        return "A"
    elif score >= 80:
        return "B"
    elif score >= 70:
        return "C"
    elif score >= 60:
        return "D"
    else:
        return "F"

try:
    score = int(input("Enter Student Score: "))
    print(calculate_grade(score))

except ValueError as e:
    print(e)
```

```
Enter Student Score: 60
D
```

Question2:

You are tasked with implementing a Python program to compute the Discrete Fourier Transform (DFT) of a given discrete signal x[n] consisting of N samples. However, you are only allowed to use basic mathematical operations and functions provided by the math module, and you must avoid relying on external libraries such as NumPy or SciPy.

Write a Python function compute_dft that takes the discrete signal x[n] as input and returns its DFT X[k], where k represents the frequency index. Your implementation should follow the formula for the DFT, which is given by:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-i2\pi kn/N}$$

Your function should implement this formula using basic mathematical operations and functions provided by the math module, such as exponentiation and trigonometric functions.

In [26]:
```
import math

def compute_dft(x):
    N = len(x)
    X = []
    for k in range(N):
        real_part = 0.0
        imag_part = 0.0
        for n in range(N):
            angle = 2 * math.pi * k * n / N
            real_part += x[n] * math.cos(angle)
            imag_part -= x[n] * math.sin(angle)
        X.append(complex(real_part, imag_part))
    return X


x = [1, 2, 3, 4]
dft_result = compute_dft(x)
for k, Xk in enumerate(dft_result):
    print(f"X[{k}] = {Xk}")
```

```
X[0] = (10+0j)
X[1] = (-2.0000000000000004+1.9999999999999996j)
X[2] = (-2-9.797174393178826e-16j)
X[3] = (-1.9999999999999982-2.000000000000001j)
```

Question3:

Suppose you are managing a website that experiences an average of 3 user visits per minute during peak hours. You are interested in analyzing the number of user visits occurring in a 5-minute interval using the Poisson distribution.

Write a Python program to simulate the number of user visits in a 5-minute interval based on the Poisson distribution with an average rate of 3 user visits per minute. Provide your program and the simulated number of user visits in a 5-minute interval.

The Poisson distribution is a discrete probability distribution that expresses the probability of a given number of events occurring in a fixed interval of time or space, assuming that these events occur with a known constant rate and are independent of the time since the last event. The Poisson distribution is characterized by a single parameter, λ, which represents the average rate of occurrence of the events.

The probability mass function (PMF) of the Poisson distribution is given by the formula:

Where:

- P(X=k) is the probability of observing k events in the interval.

- e is the base of the natural logarithm, approximately equal to 2.71828.
- λ is the average rate of occurrence of events in the interval.
- k is the number of events observed in the interval.
- k! represents the factorial of k, the product of all positive integers up to k.

This formula gives the probability of observing exactly k events in the interval, given the average rate λ. The Poisson distribution is commonly used in various fields, including queuing theory, telecommunications, biology, and finance, to model the occurrence of rare events over time or space.

In [23]:
```python
import random
import math

def simulate_user_visits(average_rate_per_minute, interval_minutes):

  # Calculate the average rate for the entire interval
  average_rate = average_rate_per_minute * interval_minutes

  # Check for Python version compatibility
  if hasattr(random, 'poissonvariate'):
    number_of_visits = random.poissonvariate(average_rate)
  else:
    # Implement Poisson PMF for older Python versions
    lambda_exp = math.exp(-average_rate)
    number_of_visits = 0
    k = 0
    while True:
      p = (average_rate**k) * lambda_exp / math.factorial(k)
      if p < 1e-30:  # Threshold for very small probabilities
        break
      if random.random() < p:
        number_of_visits += k
      k += 1

  return number_of_visits

# Example Usage
average_rate_per_minute = 3
interval_minutes = 5

simulated_visits = simulate_user_visits(average_rate_per_minute, interval_minutes)
print(f"Simulated number of user visits in {interval_minutes} minutes: {simulated_visits}")
```

```
Simulated number of user visits in 5 minutes: 12
```

Question4:

You are a manager at a retail store, and you need to efficiently manage your inventory. Your task is to categorize and organize the products in the inventory based on their popularity and availability.

You decide to use Python to assist you in this task. You have a list of products in your inventory, each represented by a tuple containing the product name, its category, and its availability status (True for available, False for out of stock).

Your goal is to:

- Enumerate through the list of products and assign a unique product ID to each product.
- Use the zip function to create a dictionary that categorizes the products based on their availability status (available or out of stock).
- Randomly select a product from the available products to promote as a special offer.

Write a Python program to accomplish the above tasks using the enumerate, zip, and random libraries.

Example of Inventory:

inventory = [("Smartphone", "Electronics", True), ("Jeans", "Apparel", True), ("Laptop", "Electronics", False), ("T-shirt", "Apparel", True), ("Headphones", "Electronics", True)]

Task 1: Enumerate through the list of products and assign a unique product ID to each product

Task 2: Use the zip function to create a dictionary that categorizes the products based on their availability status

Task 3: Randomly select a product from the available products to promote as a special offer

In [1]:
```python
import random

inventory = [
    ("Smartphone", "Electronics", True),
    ("Jeans", "Apparel", True),
    ("Laptop", "Electronics", False),
    ("T-shirt", "Apparel", True),
    ("Headphones", "Electronics", True)
]

# Task 1:
enumerated_inventory = [(idx, *product) for idx, product in enumerate(inventory, start=1)]

# Task 2:
available_products = [product for product in enumerated_inventory if product[3]]
out_of_stock_products = [product for product in enumerated_inventory if not product[3]]

categorized_products = {
    "Available": available_products,
    "Out of Stock": out_of_stock_products
}

# Task 3:
if available_products:
    special_offer = random.choice(available_products)
else:
    special_offer = None

# Output results
print("Enumerated Inventory:")
for product in enumerated_inventory:
    print(product)

print("\nCategorized Products:")
print("Available:")
for product in categorized_products["Available"]:
    print(product)
print("Out of Stock:")
for product in categorized_products["Out of Stock"]:
    print(product)

print("\nSpecial Offer Product:")
if special_offer:
    print(special_offer)
else:
    print("No available products to promote as a special offer.")
```

```
Enumerated Inventory:
(1, 'Smartphone', 'Electronics', True)
(2, 'Jeans', 'Apparel', True)
(3, 'Laptop', 'Electronics', False)
(4, 'T-shirt', 'Apparel', True)
(5, 'Headphones', 'Electronics', True)

Categorized Products:
Available:
(1, 'Smartphone', 'Electronics', True)
(2, 'Jeans', 'Apparel', True)
(4, 'T-shirt', 'Apparel', True)
(5, 'Headphones', 'Electronics', True)
Out of Stock:
(3, 'Laptop', 'Electronics', False)

Special Offer Product:
(5, 'Headphones', 'Electronics', True)
```

Question5:

You are a teacher preparing for an upcoming class quiz, and you want to create a randomized quiz generator to ensure that each student receives a unique set of questions.This will prevent cheating and encourage individual understanding of the material.

Your goal is to create a Python program that generates randomized quizzes consisting of questions from a predefined question bank.

Your program should:

- Define a question bank containing a variety of questions on different topics or subjects.
- Implement logic to randomly select a specified number of questions from the question bank for each quiz.
- Ensure that each quiz generated has a unique set of randomized questions, preventing duplicates.
- Optionally, include features such as shuffling the order of questions within each quiz and providing answer keys for grading purposes.

Write a Python program to accomplish the above tasks, allowing teachers to generate randomized quizzes quickly and efficiently for their students.

In [25]:
```python
import random

# Define a question bank
question_bank = [
    {"question": "What is the capital of Pakistan?", "answer": "Islamabad"},
    {"question": "What is 16 + 2?", "answer": "18"},
    {"question": "Who wrote 'To Kill a Mockingbird'?", "answer": "Harper Lee"},
    {"question": "What is the largest planet in our solar system?", "answer": "Jupiter"},
    {"question": "What is the boiling point of water?", "answer": "100°C"},
    {"question": "Who painted the Mona Lisa?", "answer": "Leonardo da Vinci"},
    {"question": "What is the square root of 64?", "answer": "8"},
    {"question": "Who is the author of '1984'?", "answer": "George Orwell"},
    {"question": "What is the chemical symbol for gold?", "answer": "Au"},
    {"question": "What is the tallest mountain in the world?", "answer": "Mount Everest"}
]

def generate_quiz(question_bank, num_questions):
    # Randomly select the specified number of questions from the question bank
    selected_questions = random.sample(question_bank, num_questions)
    # Shuffle the order of questions within the quiz
    random.shuffle(selected_questions)
    return selected_questions

def generate_answer_key(quiz):
    answer_key = {i+1: question["answer"] for i, question in enumerate(quiz)}
    return answer_key

# Generate a quiz for each student
num_questions = 5
num_students = 3

for student_id in range(1, num_students + 1):
    print(f"Quiz for Student {student_id}:")
    quiz = generate_quiz(question_bank, num_questions)
    for i, question in enumerate(quiz):
        print(f"{i+1}. {question['question']}")
    print()

    # Generate and print the answer key
    answer_key = generate_answer_key(quiz)
    print(f"Answer Key for Student {student_id}:")
    for q_num, answer in answer_key.items():
        print(f"{q_num}. {answer}")
    print("---------\n")
```

```
Quiz for Student 1:
1. Who wrote 'To Kill a Mockingbird'?
2. Who painted the Mona Lisa?
3. What is the capital of France?
4. What is the boiling point of water?
5. What is the square root of 64?

Answer Key for Student 1:
1. Harper Lee
2. Leonardo da Vinci
3. Paris
4. 100°C
5. 8
---------

Quiz for Student 2:
1. What is the largest planet in our solar system?
2. What is the square root of 64?
3. Who is the author of '1984'?
4. What is 2 + 2?
5. What is the capital of France?

Answer Key for Student 2:
1. Jupiter
2. 8
3. George Orwell
4. 4
5. Paris
---------

Quiz for Student 3:
1. What is the square root of 64?
2. What is the largest planet in our solar system?
3. Who is the author of '1984'?
4. What is the tallest mountain in the world?
5. Who wrote 'To Kill a Mockingbird'?

Answer Key for Student 3:
1. 8
2. Jupiter
3. George Orwell
4. Mount Everest
5. Harper Lee
---------
```