Sample Programs for Algorithm, Flowchart, and Pseudocode Development

1. Student Grade Calculator

Algorithm:-

Start

Input the assignment score (assignments)

Input the midterm exam score (midterm)

Input the final exam score (final)

Calculate the weighted average:

final_grade = (assignments * 0.30) + (midterm * 0.30) + (final * 0.40)

Determine pass/fail status:

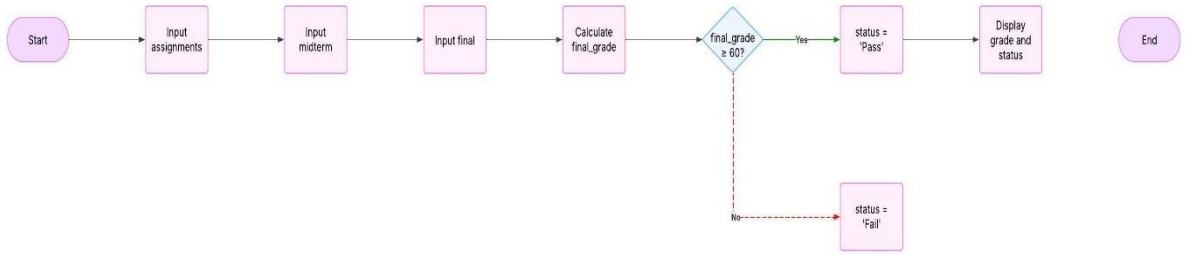
If final_grade ≥ 60, status = "Pass"

Else, status = "Fail"

Display the final grade and status

End

Flowchart:-



Pseudocode:-

```
START
  READ assignments
  READ midterm
  READ final
  final_grade = (assignments * 0.30) + (midterm * 0.30) + (final * 0.40)
  IF final_grade >= 60
    status = "Pass"
  ELSE
    status = "Fail"
  END IF
```

```
  PRINT "Final Grade:", final_grade
  PRINT "Status:", status
END
```

---

2. ATM Banking System

Algorithm:-

Start

Initialize: balance = 1000, pin = "1234"

Authenticate:

Input user_pin

If user_pin ≠ pin: Print "Incorrect PIN. Exiting..." → Exit

Loop:

Display Menu:

"1. Check Balance", "2. Deposit Money", "3. Withdraw Money", "4. Exit"

Input Choice: choice

Process Choice:

If choice = 1: Print balance

If choice = 2:

Input deposit_amount → Update balance = balance + deposit_amount → Print balance

If choice = 3:

Input withdrawal_amount → If withdrawal_amount > balance: Print "Insufficient funds."

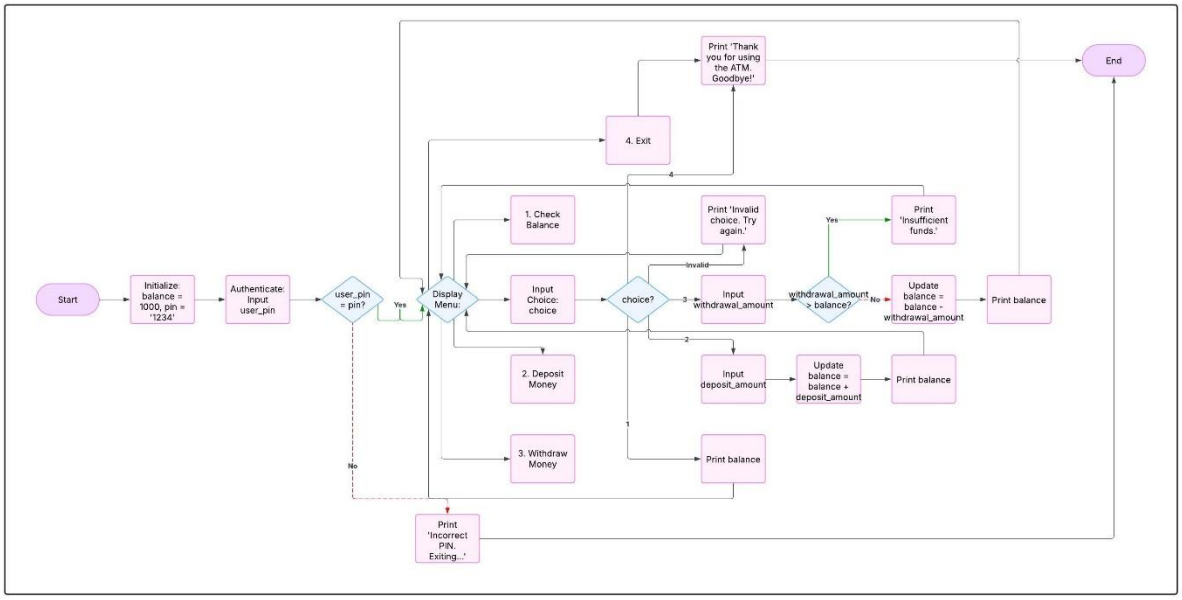
Else: Update balance = balance - withdrawal_amount → Print balance

If choice = 4: Print "Thank you for using the ATM. Goodbye!" → Exit

Else: Print "Invalid choice. Try again."

End

Flowchart:-



Pseudocode:-

START

balance = 1000

pin = "1234"

PRINT "Welcome to the ATM System"

INPUT user_pin

IF user_pin ≠ pin

PRINT "Incorrect PIN. Exiting..."

EXIT

END IF

WHILE TRUE

PRINT "Menu:"

PRINT "1. Check Balance"

PRINT "2. Deposit Money"

PRINT "3. Withdraw Money"

```
    PRINT "4. Exit"
    INPUT choice
    IF choice == 1
      PRINT "Current Balance:", balance
    ELSE IF choice == 2
      INPUT deposit_amount
      balance = balance + deposit_amount
      PRINT "Deposit successful. Updated Balance:", balance
    ELSE IF choice == 3
      INPUT withdrawal_amount
      IF withdrawal_amount > balance
        PRINT "Insufficient funds."
      ELSE
        balance = balance - withdrawal_amount
        PRINT "Withdrawal successful. Updated Balance:", balance
      END IF
    ELSE IF choice == 4
      PRINT "Thank you for using the ATM. Goodbye!"
      EXIT
    ELSE
      PRINT "Invalid choice. Please try again."
    END IF
  END WHILE
END
```

---

3. Inventory Management System

Algorithm:-

Start

Initialize inventory list

Display menu options:

Add Item

Update Item

Remove Item

Search Item

Display Inventory

Generate Report

Get user choice

If choice is "Add Item":

Input item ID, name, price, quantity

Add item to inventory

If choice is "Update Item":

Input item ID

Update item information (name, price, quantity)

If choice is "Remove Item":

Input item ID

Remove item from inventory

If choice is "Search Item":

Input item ID or name

Display item details

If choice is "Display Inventory":

Show all items in inventory

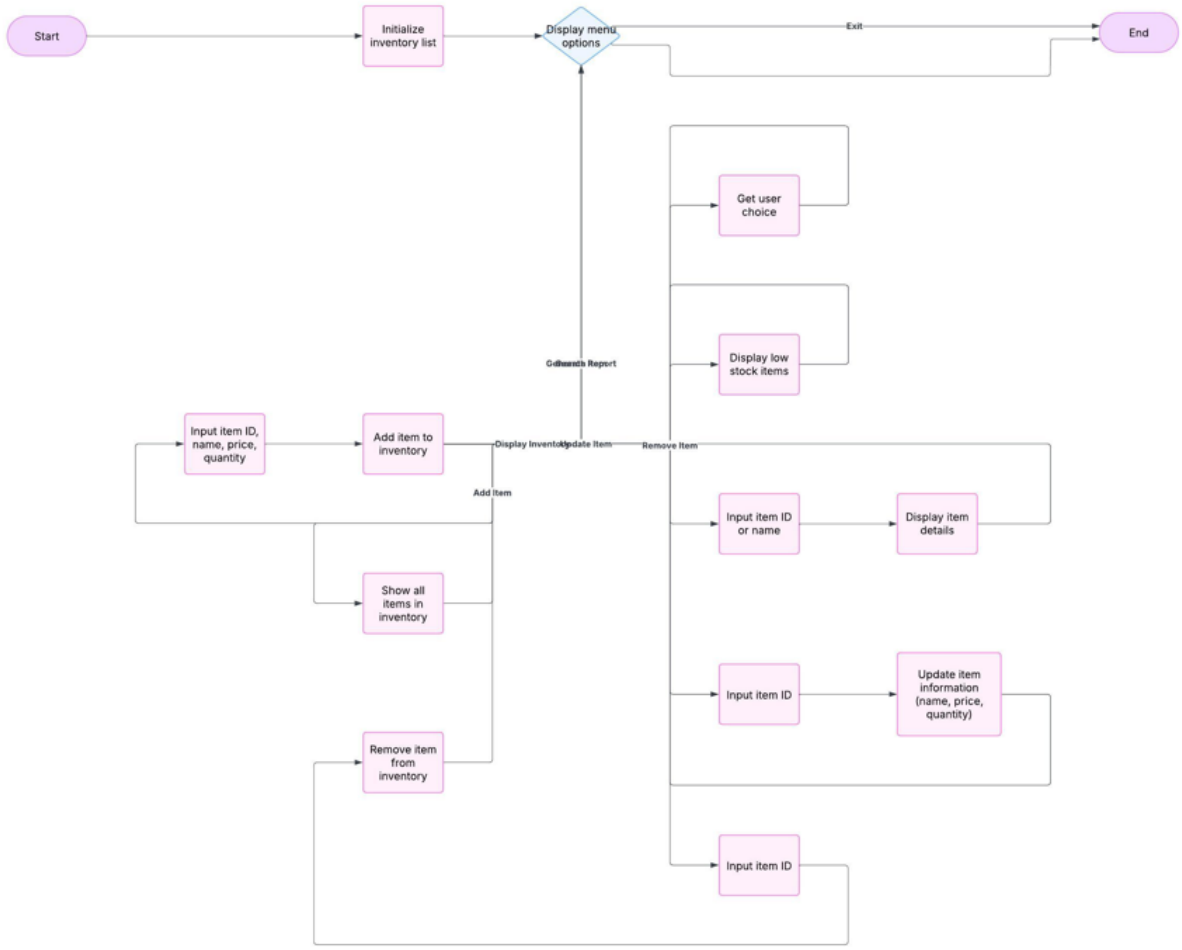
If choice is "Generate Report":

Display low stock items

Repeat from step 3 until user exits

End

Flowchart:-



Pseudocode:-

```
BEGIN
  INITIALIZE inventory_list
  WHILE true DO
    DISPLAY menu options
    GET user_choice
    IF user_choice = "Add Item" THEN
      READ item_id, item_name, item_price, item_quantity
      ADD item to inventory_list
```

```
    ELSE IF user_choice = "Update Item" THEN
      READ item_id
      UPDATE item details in inventory_list
    ELSE IF user_choice = "Remove Item" THEN
      READ item_id
      REMOVE item from inventory_list
    ELSE IF user_choice = "Search Item" THEN
      READ id_or_name
      DISPLAY item details
    ELSE IF user_choice = "Display Inventory" THEN
      SHOW all items in inventory_list
    ELSE IF user_choice = "Generate Report" THEN
      DISPLAY low stock items
    ELSE IF user_choice = "Exit" THEN
      BREAK
  END WHILE
END
```

---

4. Prime Number Checker

Algorithm:-

Start

Input num

If num <= 1: Print "Not a prime number." → End

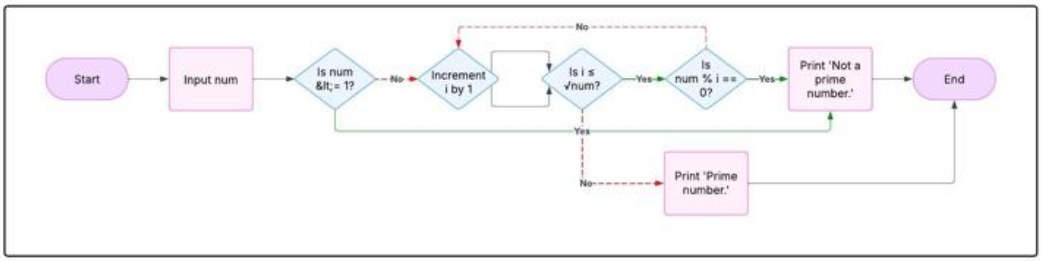
For i = 2 to √num:

If num % i == 0: Print "Not a prime number." → End

Print "Prime number."

End

Flowchart:-



Pseudocode:-

START

INPUT num

IF num <= 1

  PRINT "Not a prime number."

ELSE

  FOR i = 2 TO √num

    IF num % i == 0

      PRINT "Not a prime number."

      EXIT

  PRINT "Prime number."

END

---

5. Temperature Conversion Tool

Algorithm:-

Start

Input temperature, source_unit, target_unit

If source_unit == "C":

If target_unit == "F": result = (temperature * 9/5) + 32

If target_unit == "K": result = temperature + 273.15

If source_unit == "F":

If target_unit == "C": result = (temperature - 32) * 5/9

If target_unit == "K": result = (temperature - 32) * 5/9 + 273.15

If source_unit == "K":

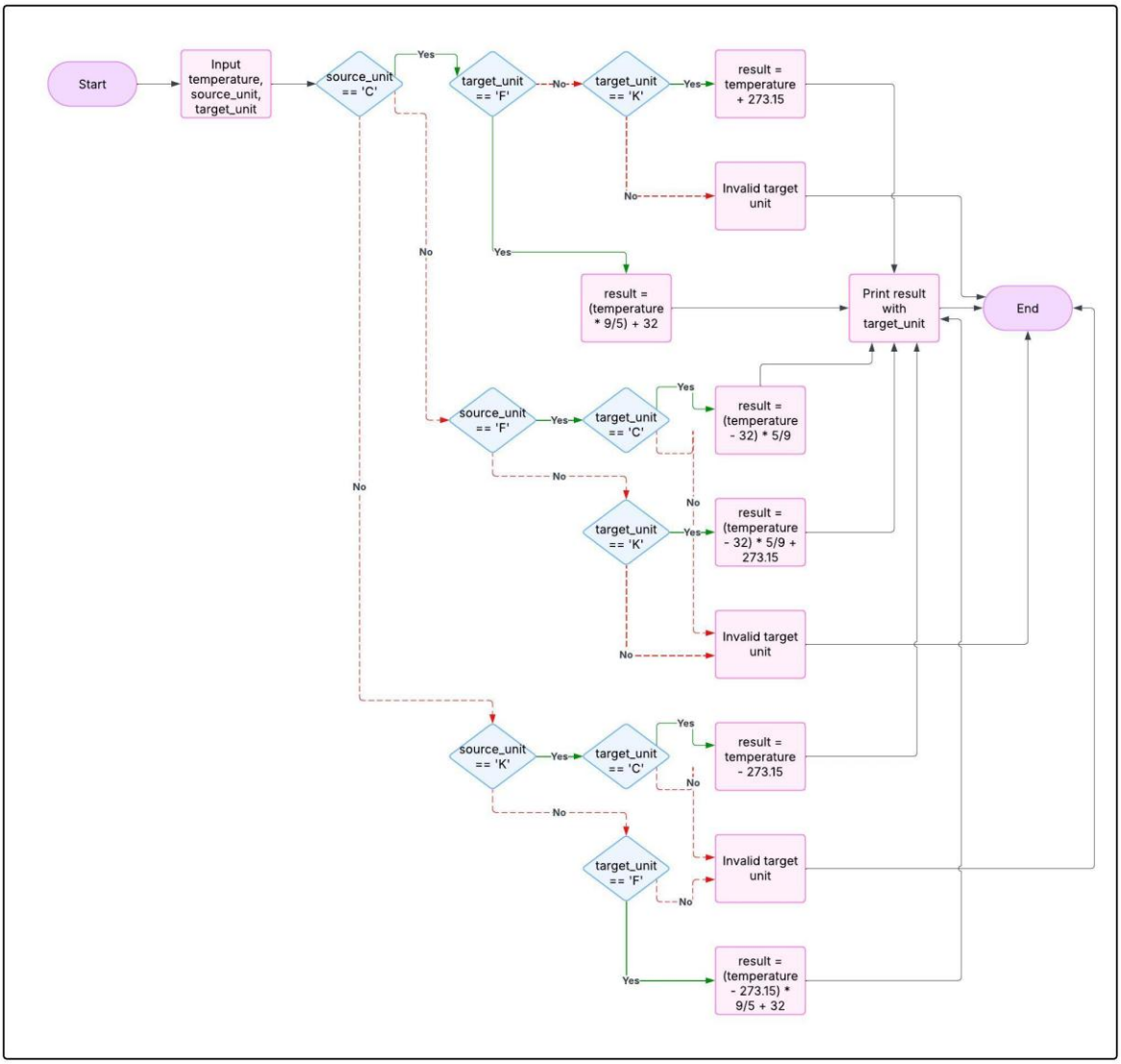
If target_unit == "C": result = temperature - 273.15

If target_unit == "F": result = (temperature - 273.15) * 9/5 + 32

Print result with target_unit

End

Flowchart:-



Pseudocode:-

```
START
INPUT temperature, source_unit, target_unit
IF source_unit == "C"
  IF target_unit == "F"
    result = (temperature * 9/5) + 32
  ELSE IF target_unit == "K"
    result = temperature + 273.15
ELSE IF source_unit == "F"
  IF target_unit == "C"
    result = (temperature - 32) * 5/9
  ELSE IF target_unit == "K"
    result = (temperature - 32) * 5/9 + 273.15
ELSE IF source_unit == "K"
  IF target_unit == "C"
    result = temperature - 273.15
  ELSE IF target_unit == "F"
    result = (temperature - 273.15) * 9/5 + 32
PRINT result, target_unit
END
```

---

6. Library Book Management System

Algorithm:-

Start

Initialize books and members databases

Display menu:

Add Book

Remove Book

Check Out Book

Return Book

Search Book

Generate Overdue Report

Perform selected operation:

Add Book: Input title, author, ISBN; add to books with status = "Available"

Remove Book: Input ISBN; remove from books

Check Out Book: Input ISBN, member_id; update status = "Checked Out" and set due_date

Return Book: Input ISBN; update status = "Available"; calculate late_fee if overdue

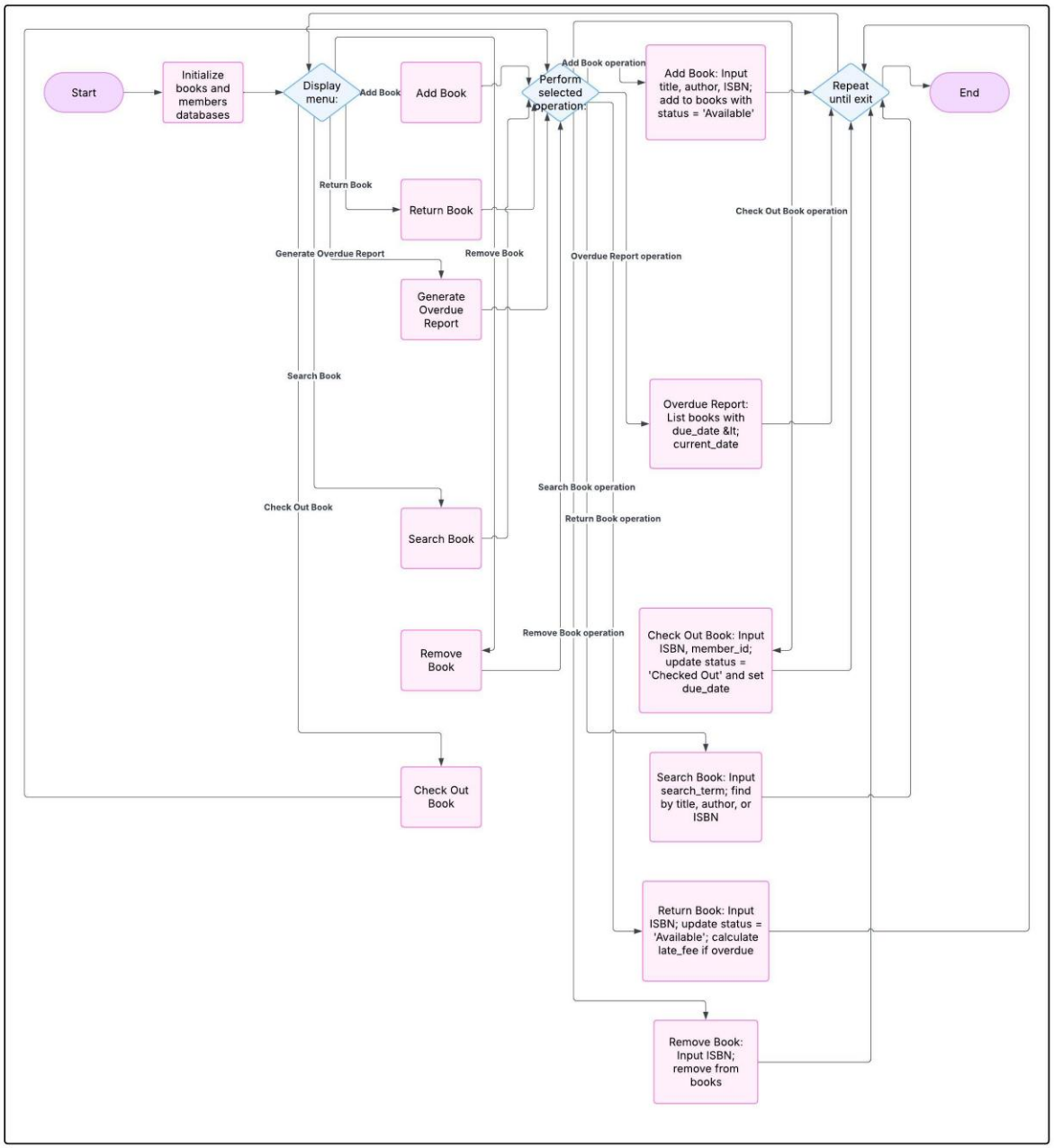
Search Book: Input search_term; find by title, author, or ISBN

Overdue Report: List books with due_date < current_date

Repeat until exit

End

Flowchart:-



Pseudocode:-

START

books = [], members = []

WHILE TRUE

PRINT "Menu: 1. Add Book, 2. Remove Book, 3. Check Out Book, 4. Return Book, 5. Search Book, 6. Overdue Report, 7. Exit"

INPUT choice

```
 IF choice == 1
   INPUT title, author, ISBN
   ADD {"title": title, "author": author, "ISBN": ISBN, "status": "Available"} TO books
 ELSE IF choice == 2
   INPUT ISBN
   REMOVE book FROM books WHERE book["ISBN"] == ISBN
 ELSE IF choice == 3
   INPUT ISBN, member_id
   FIND book IN books WHERE book["ISBN"] == ISBN
   IF book["status"] == "Available"
  UPDATE book["status"] = "Checked Out", book["due_date"] = current_date + 14
   ELSE
   PRINT "Book not available."
 ELSE IF choice == 4
   INPUT ISBN
  FIND book IN books WHERE book["ISBN"] == ISBN
 UPDATE book["status"] = "Available"
 IF book["due_date"] < current_date
CALCULATE late_fee = (current_date - book["due_date"]) * 1
PRINT "Late fee:", late_fee
 ELSE IF choice == 5
INPUT search_term
FIND book IN books WHERE book["title"] == search_term OR book["author"] ==
search_term OR book["ISBN"] == search_term
PRINT book
ELSE IF choice == 6
PRINT "Overdue Books:"
 FOR book IN books
```

IF book["due_date"] < current_date

PRINT book

ELSE IF choice == 7

EXIT

END

---

7. Fibonacci Sequence Generator

Algorithm:-

Start

Input num_terms

If num_terms <= 0: Print "Invalid input. Please enter a positive integer." → End

Initialize fib_sequence = [0, 1]

For i = 2 to num_terms - 1:

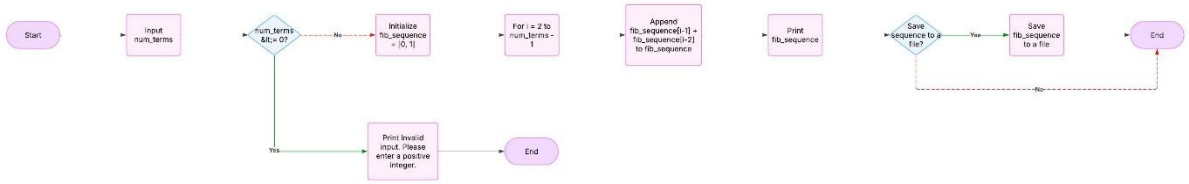
Append fib_sequence[i-1] + fib_sequence[i-2] to fib_sequence

Print fib_sequence

Optionally, save fib_sequence to a file

End

Flowchart:-



Pseudocode:-

START

INPUT num_terms

IF num_terms <= 0

  PRINT "Invalid input. Please enter a positive integer."

ELSE

```
fib_sequence = [0, 1]
FOR i = 2 TO num_terms - 1
  APPEND fib_sequence[i-1] + fib_sequence[i-2] TO fib_sequence
PRINT fib_sequence
INPUT "Save to file? (Y/N): ", save_choice
IF save_choice == "Y"
  SAVE fib_sequence TO "fibonacci_sequence.txt"
END
```

---

8. Calendar Event Scheduler

Algorithm:-

Start

Initialize events

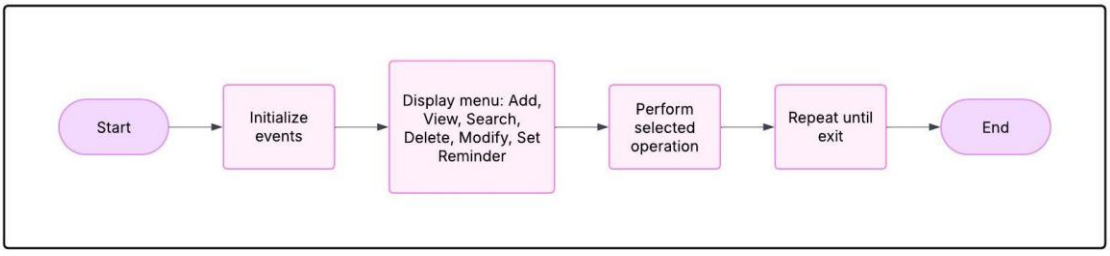
Display menu: Add, View, Search, Delete, Modify, Set Reminder

Perform selected operation

Repeat until exit

End

Flowchart:-



Pseudocode:-

```
START
events = []
WHILE TRUE
  INPUT choice
```

IF choice == 1: ADD event

ELSE IF choice == 2: VIEW events

ELSE IF choice == 3: SEARCH events

ELSE IF choice == 4: DELETE event

ELSE IF choice == 5: MODIFY event

ELSE IF choice == 6: SET reminder

ELSE IF choice == 7: EXIT

END