### **1. Overview**

The program uses arrays to store income and expense data, as well as variables for tracking savings, savings goals, and monthly budgets. Users interact with the program through a menu-driven interface in the main function.

### **2. Key Components**

#### **Constants and Arrays**

c

CopyEdit

#define MAX\_ENTRIES 100

* + This defines the maximum number of entries for income and expense categories and amounts.

c

CopyEdit

char incomeCategories[MAX\_ENTRIES][50];  
double incomeAmounts[MAX\_ENTRIES];  
int incomeCount = 0;

* + incomeCategories: Stores income category names (e.g., Salary, Bonus).
  + incomeAmounts: Stores corresponding income amounts.
  + incomeCount: Tracks the number of income entries added.

c

CopyEdit

char expenseCategories[MAX\_ENTRIES][50];  
double expenseAmounts[MAX\_ENTRIES];  
int expenseCount = 0;

* + Similarly, these arrays store expense categories, amounts, and their count.

#### **Savings, Goals, and Budget**

c

CopyEdit

double savings = 0;  
double savingsGoal = 0;  
double monthlyBudget = 0;

* + savings: Tracks current savings.
  + savingsGoal: User-defined target for savings.
  + monthlyBudget: User-defined monthly budget.

### **3. Functions**

#### **addIncome**

c

CopyEdit

void addIncome() {  
 if (incomeCount >= MAX\_ENTRIES) {  
 printf("Income entries are full!\n");  
 return;  
 }  
 printf("Enter income category: ");  
 scanf("%49s", incomeCategories[incomeCount]);  
 printf("Enter income amount: ");  
 scanf("%lf", &incomeAmounts[incomeCount]);  
 incomeCount++;  
}

* + Prompts the user to input a category (e.g., "Salary") and amount (e.g., 2000).
  + Stores this data in incomeCategories and incomeAmounts.
  + Increments incomeCount.

#### **addExpense**

c

CopyEdit

void addExpense() {  
 if (expenseCount >= MAX\_ENTRIES) {  
 printf("Expense entries are full!\n");  
 return;  
 }  
 printf("Enter expense category: ");  
 scanf("%49s", expenseCategories[expenseCount]);  
 printf("Enter expense amount: ");  
 scanf("%lf", &expenseAmounts[expenseCount]);  
 expenseCount++;  
}

* + Works similarly to addIncome but for expenses.

#### **setMonthlyBudget**

c

CopyEdit

void setMonthlyBudget() {  
 printf("Enter your monthly budget: ");  
 scanf("%lf", &monthlyBudget);  
}

* + Allows the user to define a monthly spending limit.

#### **updateSavings**

c

CopyEdit

void updateSavings() {  
 printf("Enter your current savings: ");  
 scanf("%lf", &savings);  
 printf("Enter your savings goal: ");  
 scanf("%lf", &savingsGoal);  
}

* + Updates the savings and savingsGoal variables.

#### **viewReport**

c

CopyEdit

void viewReport() {  
 double totalIncome = 0, totalExpenses = 0;  
  
 printf("\n--- Financial Report ---\n");  
 printf("\nIncome:\n");  
 for (int i = 0; i < incomeCount; i++) {  
 printf("%s: %.2f\n", incomeCategories[i], incomeAmounts[i]);  
 totalIncome += incomeAmounts[i];  
 }  
 printf("Total Income: %.2f\n", totalIncome);  
  
 printf("\nExpenses:\n");  
 for (int i = 0; i < expenseCount; i++) {  
 printf("%s: %.2f\n", expenseCategories[i], expenseAmounts[i]);  
 totalExpenses += expenseAmounts[i];  
 }  
 printf("Total Expenses: %.2f\n", totalExpenses);  
  
 printf("\nSavings: %.2f\n", savings);  
 printf("Net Balance: %.2f\n", totalIncome - totalExpenses);  
  
 if (totalIncome > 0)  
 printf("Savings Rate: %.2f%%\n", (savings / totalIncome) \* 100);  
 if (monthlyBudget > 0) {  
 printf("Monthly Budget: %.2f\n", monthlyBudget);  
 printf("Remaining Budget: %.2f\n", monthlyBudget - totalExpenses);  
 }  
 if (savingsGoal > 0)  
 printf("Savings Goal: %.2f\nProgress Toward Goal: %.2f%%\n", savingsGoal, (savings / savingsGoal) \* 100);  
}

* + Calculates totalIncome and totalExpenses by summing up their respective arrays.
  + Displays:
    - Income categories and amounts.
    - Expense categories and amounts.
    - Total income, expenses, and net balance (Income - Expenses).
    - Savings rate, monthly budget, remaining budget, and progress toward savings goal.

#### **visualizeData**

c

CopyEdit

void visualizeData() {  
 printf("\n--- Data Visualization ---\n");  
  
 double totalIncome = 0, totalExpenses = 0;  
 for (int i = 0; i < incomeCount; i++)  
 totalIncome += incomeAmounts[i];  
 for (int i = 0; i < expenseCount; i++)  
 totalExpenses += expenseAmounts[i];  
  
 printf("Income: ");  
 for (int i = 0; i < (int)(totalIncome / 100); i++)  
 printf("|");  
 printf(" %.2f\n", totalIncome);  
  
 printf("Expenses: ");  
 for (int i = 0; i < (int)(totalExpenses / 100); i++)  
 printf("|");  
 printf(" %.2f\n", totalExpenses);  
  
 printf("Savings: ");  
 for (int i = 0; i < (int)(savings / 100); i++)  
 printf("|");  
 printf(" %.2f\n", savings);  
}

* + Provides a simple text-based bar chart for income, expenses, and savings.

#### **Main Menu**

c

CopyEdit

int main() {  
 int choice;  
  
 do {  
 printf("\n--- Finance Management System ---\n");  
 printf("1. Add Income\n");  
 printf("2. Add Expense\n");  
 printf("3. View Report\n");  
 printf("4. Set Monthly Budget\n");  
 printf("5. Update Savings\n");  
 printf("6. Visualize Data\n");  
 printf("7. Exit\n");  
 printf("Enter your choice: ");  
 scanf("%d", &choice);  
  
 switch (choice) {  
 case 1:  
 addIncome();  
 break;  
 case 2:  
 addExpense();  
 break;  
 case 3:  
 viewReport();  
 break;  
 case 4:  
 setMonthlyBudget();  
 break;  
 case 5:  
 updateSavings();  
 break;  
 case 6:  
 visualizeData();  
 break;  
 case 7:  
 printf("Exiting program...\n");  
 break;  
 default:  
 printf("Invalid choice. Try again.\n");  
 }  
 } while (choice != 7);  
  
 return 0;  
}

* + The main menu allows users to:
    - Add income.
    - Add expenses.
    - View the financial report.
    - Set a budget.
    - Update savings and goals.
    - Visualize data.
    - Exit the program.
  + Executes the corresponding function based on user choice.

### **4. Summary**

This program:

* 1. Provides tools for tracking financial data.
  2. Summarizes data with reports and simple visualizations.
  3. Encourages goal setting and budgeting.

It's a compact and effective tool for personal finance management!

4o