#include <stdio.h>

#define BRANCHES 3

#define MONTHS 12

void enterSalesData(float sales[BRANCHES][MONTHS]) {

for (int i = 0; i < BRANCHES; i++) { printf("Enter sales for Branch %d:\n", i + 1); for (int j = 0; j < MONTHS; j++) { printf(" Month %d: ", j + 1); scanf("%f", &sales[i][j]);

}

}

}

void calculateTotalSales(float sales[BRANCHES][MONTHS]) { float total = 0; for (int i = 0; i < BRANCHES; i++) { for (int j = 0; j < MONTHS; j++) { total += sales[i][j];

}

}

printf("Total sales of the company: %.2f\n", total);

}

void calculatePercentageShare(float sales[BRANCHES][MONTHS]) { float total = 0; for (int i = 0; i < BRANCHES; i++) { for (int j = 0; j < MONTHS; j++) { total += sales[i][j];

}

}

for (int i = 0; i < BRANCHES; i++) { float branchTotal = 0; for (int j = 0; j < MONTHS; j++) { branchTotal += sales[i][j];

}

printf("Branch %d contributes %.2f%% of total sales\n", i + 1, (branchTotal / total) \* 100);

}

}

void findPeakSalesMonth(float sales[BRANCHES][MONTHS]) { int peakMonth = 0; float peakSales = 0; for (int j = 0; j < MONTHS; j++) { float monthTotal = 0; for (int i = 0; i < BRANCHES; i++) { monthTotal += sales[i][j];

}

if (monthTotal > peakSales) { peakSales = monthTotal; peakMonth = j;

}

}

printf("Month %d had the highest sales: %.2f\n", peakMonth + 1, peakSales);

}

void sortBranchesByMonth(float sales[BRANCHES][MONTHS], int month) { float branchSales[BRANCHES]; for (int i = 0; i < BRANCHES; i++) { branchSales[i] = sales[i][month];

}

for (int i = 0; i < BRANCHES - 1; i++) { for (int j = 0; j < BRANCHES - i - 1; j++) { if (branchSales[j] < branchSales[j + 1]) { float temp = branchSales[j]; branchSales[j] = branchSales[j + 1]; branchSales[j + 1] = temp;

}

}

}

printf("Branches sorted by sales in Month %d:\n", month + 1); for (int i = 0; i < BRANCHES; i++) { printf(" Branch %d: %.2f\n", i + 1, branchSales[i]);

}

}

void sortMonthsByBranch(float sales[BRANCHES][MONTHS], int branch) { float monthSales[MONTHS];

for (int i = 0; i < MONTHS; i++) { monthSales[i] = sales[branch][i];

}

for (int i = 0; i < MONTHS - 1; i++) { for (int j = 0; j < MONTHS - i - 1; j++) { if (monthSales[j] < monthSales[j + 1]) { float temp = monthSales[j]; monthSales[j] = monthSales[j + 1]; monthSales[j + 1] = temp;

}

}

}

printf("Months sorted by sales for Branch %d:\n", branch + 1); for (int i = 0; i < MONTHS; i++) { printf(" Month %d: %.2f\n", i + 1, monthSales[i]);

}

}

int main() { float sales[BRANCHES][MONTHS]; int choice, branch, month; char continueFlag;

do {

printf("\nMenu:\n"); printf("1. Enter sales data\n"); printf("2. Calculate total sales\n"); printf("3. Calculate percentage share of branches\n"); printf("4. Find peak sales month\n"); printf("5. Sort branches by sales in a specific month\n"); printf("6. Sort months by sales for a specific branch\n"); printf("7. Exit\n"); printf("Enter your choice: "); scanf("%d", &choice);

switch (choice) { case 1:

enterSalesData(sales);

break; case 2:

calculateTotalSales(sales);

break; case 3:

calculatePercentageShare(sales);

break; case 4:

findPeakSalesMonth(sales);

break; case 5:

printf("Enter the month (1-12): ");

scanf("%d", &month); sortBranchesByMonth(sales, month - 1);

break; case 6:

printf("Enter the branch (1-%d): ", BRANCHES); scanf("%d", &branch); sortMonthsByBranch(sales, branch - 1);

break; case 7:

printf("Exiting program.\n"); return 0; default:

printf("Invalid choice. Try again.\n");

}

printf("Do you want to continue? (y/n): "); scanf(" %c", &continueFlag);

} while (continueFlag == 'y' || continueFlag == 'Y');

return 0;

}