#include <iostream>

#include <string>

using namespace std;

struct Task {

string description;

int priority;

};

// Function to perform selection sort on tasks (sorted by priority)

void selectionSort(Task tasks[], int size) {

for (int i = 0; i < size - 1; i++) {

int minIndex = i;

for (int j = i + 1; j < size; j++) {

if (tasks[j].priority < tasks[minIndex].priority) {

minIndex = j;

}

}

swap(tasks[i], tasks[minIndex]);

}

}

// Function to display tasks

void displayTasks(Task tasks[], int size) {

if (size == 0) {

cout << "No tasks available." << endl;

return;

}

cout << "\nTasks sorted by priority:" << endl;

for (int i = 0; i < size; i++) {

cout << i + 1 << ". [" << tasks[i].priority << "] " << tasks[i].description << endl;

}

}

// Function to take user input with validation (for priority)

int getValidPriority() {

int num;

while (true) {

cout << "Enter task priority (1-100, lower means higher priority): ";

cin >> num;

if (cin.fail() || num < 1 || num > 100) {

cout << "Invalid input! Please enter a number between 1 and 100.\n";

cin.clear();

while (cin.get() != '\n'); // Discard invalid input

} else {

return num;

}

}

}

// Main function (Menu-driven)

int main() {

Task tasks[100]; // Array to store tasks (max 100)

int size = 0;

int choice;

do {

cout << "\nTask Manager - Menu";

cout << "\n1. Add Task";

cout << "\n2. Sort Tasks by Priority";

cout << "\n3. Display Tasks";

cout << "\n4. Exit";

cout << "\nEnter your choice: ";

cin >> choice;

// Input validation for choice

if (cin.fail() || choice < 1 || choice > 4) {

cout << "Invalid choice! Please enter a number between 1 and 4.\n";

cin.clear();

while (cin.get() != '\n'); // Discard invalid input

continue;

}

switch (choice) {

case 1:

if (size >= 100) {

cout << "Task list is full!" << endl;

} else {

cin.ignore(); // Clear buffer before taking string input

cout << "Enter task description: ";

getline(cin, tasks[size].description);

tasks[size].priority = getValidPriority();

size++;

}

break;

case 2:

if (size == 0) {

cout << "No tasks to sort!" << endl;

} else {

selectionSort(tasks, size);

cout << "Tasks sorted successfully!" << endl;

}

break;

case 3:

displayTasks(tasks, size);

break;

case 4:

cout << "Exiting Task Manager..." << endl;

break;

}

} while (choice != 4);

return 0;

}