#include <iostream>

using namespace std;

// Оголошення структури Date

struct Date {

int hours;

int minutes;

};

// Функція для коригування часу, щоб години були в межах від 0 до 23, а хвилини від 0 до 59

void adjustTime(Date &date) {

// Якщо хвилини більше 59, коригуємо години та хвилини

while (date.minutes >= 60) {

date.minutes -= 60;

date.hours++;

}

// Якщо години більше 23, коригуємо години

while (date.hours >= 24) {

date.hours -= 24;

}

}

// Функція для виведення часу у форматі ГГ:ХХ

void printTime(const Date &date) {

cout << date.hours << ":" << (date.minutes < 10 ? "0" : "") << date.minutes << endl;

}

int main() {

Date startTime;

int additionalMinutes;

// Введення початкового часу

cout << "Введіть години початкового часу: ";

cin >> startTime.hours;

cout << "Введіть хвилини початкового часу: ";

cin >> startTime.minutes;

// Введення кількості хвилин, які треба додати

cout << "Введіть кількість хвилин для додавання: ";

cin >> additionalMinutes;

// Додавання хвилин

startTime.minutes += additionalMinutes;

// Коригуємо час

adjustTime(startTime);

// Виведення результату

printTime(startTime);

return 0;

}

#include <iostream>

using namespace std;

// Структура для зберігання часу

struct Date {

int hours;

int minutes;

// Присвоєння значень за замовчуванням

Date(int h = 0, int m = 0) : hours(h), minutes(m) {}

};

// Функція для коригування часу, щоб години були в межах від 0 до 23, а хвилини від 0 до 59

void adjustTime(Date &date) {

// Якщо хвилини більше 59, коригуємо години та хвилини

while (date.minutes >= 60) {

date.minutes -= 60;

date.hours++;

}

// Якщо години більше 23, коригуємо години

while (date.hours >= 24) {

date.hours -= 24;

}

}

// Функція для обчислення різниці часу

Date calculateDuration(const Date &startTime, const Date &endTime) {

int startTotalMinutes = startTime.hours \* 60 + startTime.minutes;

int endTotalMinutes = endTime.hours \* 60 + endTime.minutes;

// Якщо час завершення менший за час початку, то подія тривала через добу

if (endTotalMinutes < startTotalMinutes) {

endTotalMinutes += 24 \* 60; // Додаємо 24 години в хвилинах

}

int durationMinutes = endTotalMinutes - startTotalMinutes;

// Обчислюємо години і хвилини тривалості

int durationHours = durationMinutes / 60;

int durationMins = durationMinutes % 60;

return Date(durationHours, durationMins);

}

// Функція для виведення часу у форматі ГГ:ХХ

void printTime(const Date &date) {

cout << date.hours << ":" << (date.minutes < 10 ? "0" : "") << date.minutes << endl;

}

int main() {

Date startTime, endTime;

// Введення часу початку події

cout << "Введіть час початку події (години хвилини): ";

cin >> startTime.hours >> startTime.minutes;

// Введення часу завершення події

cout << "Введіть час завершення події (години хвилини): ";

cin >> endTime.hours >> endTime.minutes;

// Обчислення тривалості події

Date duration = calculateDuration(startTime, endTime);

// Виведення тривалості події

cout << "Тривалість події: ";

printTime(duration);

return 0;

}