

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Національний аерокосмічний університет ім. М. Є. Жуковського  
«Харківський авіаційний інститут»

Факультет систем управління літальних апаратів  
Кафедра систем управління літальних апаратів

## **Лабораторна робота № 10**

з дисципліни «Алгоритмізація та програмування»  
на тему «Створення і обробка структур даних мовою C ++»

XAI.301.175.318.16ЛР

Виконав студент гр. 318

Нікуліна К.О.  
(підпис, дата) (П.І.Б.)

Перевірів

                     к.т.н., доц. Олена ГАВРИЛЕНКО  
(підпис, дата) (П.І.Б.)

## МЕТА РОБОТИ

Вивчити теоретичний матеріал з основ представлення структур (записів) мовою C ++, а також їх передачі в функції, і реалізувати декларування і обробку структур мовою C ++ в середовищі Visual Studio.

## ПОСТАНОВКА ЗАДАЧІ

Завдання 1. Вирішити задачу зі структурами даних. Варіанти завдань представлено на рис.1.

Використовуючи тип TTime (див. Param71), описати процедуру NextSec (T) з параметром типу TTime, яка змінює час на +1 секунду (якщо час T є неправильним, то воно не змінюється). Запис T є вхідним і вихідним параметром. Застосувати процедуру NextSec до п'яти заданих моментів часу.

Рис.1

Завдання 2. Для задач з рис.2-3:

А. Описати структуру, яка містить всі вхідні і всі вихідні дані задачі.

В. Визначити функцію (\*метод), що реалізує обробку структури відповідно до задачі.

С. Визначити функцію (\*метод), що перевіряє на коректність і заповнює відповідні поля вхідних даних структури

Д. Викликати функції (\*методи) з пунктів С, В після оголошення змінної (об'єкту) структури.

Е. Вивести значення полів вихідних даних.

---

Поміняти місцями вміст змінних А і В і вивести нові значення А і В.

Рис.2

Дано цілі числа a, b, c. Перевірити істинність висловлювання: «Існує трикутник зі сторонами a, b, c».

Рис.3

Завдання 3. Рішення всіх трьох задач реалізувати в одному консольному додатку, \*структурувати на модулі.

## ВИКОНАННЯ РОБОТИ

Завдання 1.

Вирішення задачі Param74

Вхідні дані (ім'я, опис, тип, обмеження):

TTime – час, виражений у форматі: час, хвилини, секунди;

- Дійсний тип;
- Обмеження:
  - Години:  $0 \leq \text{hours} \leq 23$ ;
  - Хвилини:  $0 \leq \text{minutes} \leq 59$ ;
  - Секунди:  $0 \leq \text{seconds} \leq 59$ .

Вихідні дані (ім'я, опис, тип):

T - час зменшено на 1 секунду, дійсний тип.

Алгоритм вирішення:

1. Перевірка коректності часу (якщо не входить у обмеження - завершити програму);
2. Зменшення на 1 секунду:
  - Якщо значення секунд приймає значення “-1”:
    - Значення секунд отримує 59;
    - Значення хвилини змінюється на “-1” від даної хвилини-отримує значення 59:
      - Якщо значення години приймає значення “-1” - значення години отримує 23.
3. Вивід результатів.

Завдання 2.

Вирішення задачі Begin22

Вхідні дані (ім'я, опис, тип, обмеження):

1. A – число, дійсний тип;
2. B – число, дійсний тип.

Вихідні дані (ім'я, опис, тип):

1. A – число, дійсний тип;
2. B – число, дійсний тип.

Алгоритм вирішення:

1. Використати арифметичні операції:

- $A = A + B$ ,
- $B = A - B$ ,
- $A = A - B$ .

2. Вивід результатів.

Завдання 3.

Вирішення задачі Boolean33

Вхідні дані (ім'я, опис, тип, обмеження):

1. a – число, дійсний тип;
2. b – число, дійсний тип;
3. c – число, дійсний тип;

Вихідні дані (ім'я, опис, тип):

res – логічна змінна перевіряюча, що «Існує трикутник зі сторонами a, b, c».

Алгоритм вирішення:

1. Перевірка на сторін на від'ємність;
2. Вивід результату.

Лістинг коду вирішення задачі розділ і номер задач Param76, Begin22, Boolean33 наведено в дод. А (стор. 6). Екран роботи програми показаний на рис. Б.1.

## ВИСНОВКИ

Вивчений теоретичний матеріал з основ уявлення, передач у функції, оголошення, та обробки структур.

## ДОДАТОК А

### Лістинг коду програми

```
#include <iostream>
#include <iomanip>

using namespace std;

// Структура для задания 1
struct TTime {
    int Hour;    // 0..23
    int Min;     // 0..59
    int Sec;     // 0..59
    bool IsCorrect;
};

// Прототипы функций для задания 1
void ShowTime(const TTime& T);
bool IsValidTime(int Hour, int Min, int Sec);
void SetTime(TTime& T, int Hour, int Min, int Sec);
void NextSec(TTime& T);
void Task1();

// Прототип функции для задания 2
void Task2();

// Прототип функции для задания 3
void Task3();

int main()
{
    setlocale(LC_ALL, "Russian");

    int choice;
    bool exitProgram = false;

    while (!exitProgram)
    {
        cout << "\n=== МЕНЮ ===" << endl;
        cout << "1. Задание 1: Процедура NextSec (изменение времени)" << endl;
        cout << "2. Задание 2: Обмен значений переменных А и В" << endl;
        cout << "3. Задание 3: Проверка существования треугольника" << endl;
        cout << "0. Выход" << endl;
```

```

    cout << "Введите номер задания: ";
    cin >> choice;

    switch (choice)
    {
    case 1:
        Task1();
        break;
    case 2:
        Task2();
        break;
    case 3:
        Task3();
        break;
    case 0:
        exitProgram = true;
        break;
    default:
        cout << "Некорректный выбор. Попробуйте снова." << endl;
    }
}

return 0;
}

//===== Задание 1: Процедура NextSec =====

void ShowTime(const TTime& T) {
    if (T.IsCorrect) {
        cout << setfill('0') << setw(2) << T.Hour << ":"
              << setfill('0') << setw(2) << T.Min << ":"
              << setfill('0') << setw(2) << T.Sec;
    }
    else {
        cout << "Неправильное время";
    }
}

bool IsValidTime(int Hour, int Min, int Sec) {
    return (Hour >= 0) && (Hour <= 23) &&
           (Min >= 0) && (Min <= 59) &&
           (Sec >= 0) && (Sec <= 59);
}

void SetTime(TTime& T, int Hour, int Min, int Sec) {
    if (IsValidTime(Hour, Min, Sec)) {
        T.Hour = Hour;
    }
}

```

```

        T.Min = Min;
        T.Sec = Sec;
        T.IsCorrect = true;
    }
    else {
        T.IsCorrect = false;
    }
}

void NextSec(TTime& T) {
    if (!T.IsCorrect) {
        return; // Якщо час неправильний, не змінюємо його
    }

    T.Sec = T.Sec + 1;

    if (T.Sec == 60) {
        T.Sec = 0;
        T.Min = T.Min + 1;

        if (T.Min == 60) {
            T.Min = 0;
            T.Hour = T.Hour + 1;

            if (T.Hour == 24) {
                T.Hour = 0;
            }
        }
    }
}

void Task1() {
    cout << "\n=== Задание 1: Процедура NextSec ===\n" << endl;

    TTime times[5]; // Массив для хранения 5 моментов времени
    int hour, min, sec;

    // Ввод 5 моментов времени
    for (int i = 0; i < 5; i++) {
        cout << "Введите время T" << (i + 1) << " (часы минуты секунды): ";
        cin >> hour >> min >> sec;
        SetTime(times[i], hour, min, sec);
    }

    // Выводим исходные значения времени
    cout << "\nВыходные значения времени:" << endl;
}

```



```

for (int i = 0; i < 5; i++) {
    cout << "T" << (i + 1) << " = ";
    ShowTime(times[i]);
    cout << endl;
}

// Применяем процедуру NextSec к каждому моменту времени
cout << "\nПосле выполнения NextSec:" << endl;
for (int i = 0; i < 5; i++) {
    NextSec(times[i]);
    cout << "T" << (i + 1) << " = ";
    ShowTime(times[i]);
    cout << endl;
}
}

//===== Задание 2: Обмен значений переменных A и B =====

void Task2() {
    cout << "\n=== Задание 2: Обмен значений переменных A и B ===\n" << endl;

    int A, B;

    // Ввод значений A и B
    cout << "Введите значение A: ";
    cin >> A;

    cout << "Введите значение B: ";
    cin >> B;

    // Выводим начальные значения
    cout << "Начальные значения:" << endl;
    cout << "A = " << A << endl;
    cout << "B = " << B << endl;

    // Метод 1: Использование временной переменной
    int temp = A;
    A = B;
    B = temp;

    // Выводим новые значения после обмена
    cout << "\nНовые значения после обмена:" << endl;
    cout << "A = " << A << endl;
    cout << "B = " << B << endl;
}

```

```
//===== Задание 3: Проверка существования треугольника =====
```

```
void Task3() {
    cout << "\n=== Задание 3: Проверка существования треугольника ===\n" <<
endl;

    int a, b, c;

    // Введения даних
    cout << "Введите длину стороны a: ";
    cin >> a;

    cout << "Введите длину стороны b: ";
    cin >> b;

    cout << "Введите длину стороны c: ";
    cin >> c;

    // Перевірка існування трикутника за нерівністю трикутника
    bool exists = false;

    // Всі сторони мають бути додатними
    if (a > 0 && b > 0 && c > 0) {
        // Кожна сторона має бути меншою за суму двох інших
        if (a < b + c && b < a + c && c < a + b) {
            exists = true;
        }
    }

    // Виведення результату
    if (exists) {
        cout << "Высказывание истинно: треугольник со сторонами"
            << a << ", " << b << ", " << c << " существует." << endl;
    }
    else {
        cout << "Высказывание ошибочно: треугольник со сторонами"
            << a << ", " << b << ", " << c << " не существует." << endl;
    }
}
```

## ДОДАТОК Б

### Скріншот вікна виконання програми

```

=== МЕНЮ ===
1. Задание 1: Процедура NextSec (изменение времени)
2. Задание 2: Обмен значений переменных A и B
3. Задание 3: Проверка существования треугольника
0. Выход
Введите номер задания: 1

=== Задание 1: Процедура NextSec ===

Введите время T1 (часы минуты секунды): 1 2 3
Введите время T2 (часы минуты секунды): 10 20 30
Введите время T3 (часы минуты секунды): 23 59 59
Введите время T4 (часы минуты секунды): 24 60 60
Введите время T5 (часы минуты секунды): -1 -2 -3

Выходные значения времени:
T1 = 01:02:03
T2 = 10:20:30
T3 = 23:59:59
T4 = Неправильное время
T5 = Неправильное время

После выполнения NextSec:
T1 = 01:02:04
T2 = 10:20:31
T3 = 00:00:00
T4 = Неправильное время
T5 = Неправильное время

=== МЕНЮ ===
1. Задание 1: Процедура NextSec (изменение времени)
2. Задание 2: Обмен значений переменных A и B
3. Задание 3: Проверка существования треугольника
0. Выход
Введите номер задания: 2

=== Задание 2: Обмен значений переменных A и B ===

Введите значение A: 10
Введите значение B: 20
Начальные значения:
A = 10
B = 20

Новые значения после обмена:
A = 20
B = 10

=== МЕНЮ ===
1. Задание 1: Процедура NextSec (изменение времени)
2. Задание 2: Обмен значений переменных A и B
3. Задание 3: Проверка существования треугольника
0. Выход
Введите номер задания: 3

=== Задание 3: Проверка существования треугольника ===

Введите длину стороны a: 1
Введите длину стороны b: 2
Введите длину стороны c: 3
Высказывание ошибочно: треугольник со сторонами 1, 2, 3 не существует.

=== МЕНЮ ===
1. Задание 1: Процедура NextSec (изменение времени)
2. Задание 2: Обмен значений переменных A и B
3. Задание 3: Проверка существования треугольника
0. Выход
Введите номер задания: 3

=== Задание 3: Проверка существования треугольника ===

Введите длину стороны a: -1
Введите длину стороны b: -2
Введите длину стороны c: -3
Высказывание ошибочно: треугольник со сторонами -1, -2, -3 не существует.

=== МЕНЮ ===
1. Задание 1: Процедура NextSec (изменение времени)
2. Задание 2: Обмен значений переменных A и B
3. Задание 3: Проверка существования треугольника
0. Выход
Введите номер задания: 0

```

Рисунок Б.1 – Екран виконання програми для вирішення завдання Param76, Begin22 і Boolean33.