

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.301.174.312.8

Виконав студент гр. 312

\_\_\_\_\_ Діхтяренко М.Г.  
(підпис, дата) (П.І.Б.)

Перевірив

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

2024

## 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. Param79. Використовуючи тип TTime (див. Param71), описати функцію AddSec (T, N) типу TTime з двома входними параметрами типу TTime і цілого, яка змінює час T на + N секунд (якщо час T є неправильним, то воно повертається без змін) . За допомогою функції AddSec вивести новий час для п'яти заданих моментів часу.

Завдання 2.

Begin22. Поміняти місцями вміст змінних A і B і вивести нові значення A і B.

Boolean12. Дано три цілих числа: A, B, C. Перевірити істинність висловлювання: «Кожне з чисел A, B, C позитивне».

A. Описати структуру, яка містить всі входні і всі вихідні дані задачі.

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

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

D. Викликати функції (\*методи) з пунктів C, B після оголошення змінної (об'єкту) структури.

E. Вивести значення полів вихідних даних.

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

## ВИКОНАННЯ РОБОТИ

### Завдання 1.

#### Вирішення задачі Param79.

Типи даних:

**TTime:** Це тип даних, що представляє момент часу з годинами, хвилинами і секундами.

**hrs:** Ціле число, яке представляє години. Обмеження від 0 до 23.

**min:** Ціле число, яке представляє хвилини. Обмеження від 0 до 59.

**sec:** Ціле число, яке представляє секунди. Обмеження від 0 до 59.

#### **Обмеження:**

- Години повинні бути в межах від 0 до 23.
- Хвилини та секунди повинні бути в межах від 0 до 59.

#### Алгоритм вирішення:

Створити структуру даних для представлення моменту часу (години, хвилини, секунди).

Для вирішення завдання з редагуванням часу потрібно виконати наступні кроки:

1. Створити структуру даних для представлення моменту часу (години, хвилини, секунди).
2. Зчитати введені користувачем значення годин, хвилин та секунд.
3. Перевірити чи введені значення годин, хвилин та секунд є в межах припустимих обмежень (години від 0 до 23, хвилини та секунди від 0 до 59).
4. Зчитати введені користувачем значення кількості секунд, які потрібно додати до часу.
5. Додати цю кількість секунд до поточного часу.
6. Врахувати можливі переходи через границі (наприклад, якщо після додавання секунд кількість хвилин перевищує 59, необхідно збільшити кількість годин та відповідно зменшити кількість хвилин).
7. Вивести новий час після додавання заданої кількості секунд.

Лістинг коду вирішення задачі 1, Matrix40 наведено в дод. А (стор. 6).

Екран роботи програми наведено в дод. Б (стор. 10).

#### **Завдання 2.**

##### **Вирішення задачі Begin22.**

Тип даних:

**VVariables (структура для змінних A і B):**

- **Опис:** Ця структура буде містити дві змінні, A і B (int - цілі числа).

**Опис функції Reverse:** Ця функція приймає структуру VVariables та обмінює значеннями змінних A і B.

#### Алгоритм вирішення:

1. Зчитати значення змінних A та B з користувацького вводу.
2. Зберегти початкові значення A та B.

3. Обміняти значення змінних A і B, використовуючи тимчасову змінну для збереження одного зі значень.
4. Вивести нові значення змінних A і B.

Лістинг коду вирішення задачі 2, Begin22 наведено в дод. А (стор. 6).  
Екран роботи програми наведено в дод. Б (стор. 10).

### **Вирішення задачі Boolean12.**

Типи даних:

**NNums** (структура для чисел A, B та C):

**Опис:** Ця структура містить три цілих числа: A, B та C.

**Опис функції isPositive:** Ця функція приймає три цілих числа і повертає логічне значення true, якщо кожне з цих чисел є позитивним, та false в іншому випадку.

Алгоритм вирішення:

1. Зчитати три цілі числа з користувацького вводу (A, B та C).
2. Перевірити, чи кожне з цих чисел є позитивним (більше за 0).
3. Якщо кожне з чисел є позитивним, вивести логічне значення true, в іншому випадку - false.

Лістинг коду вирішення задачі 2, Boolean12 наведено в дод. А (стор. 6).  
Екран роботи програми наведено в дод. Б (стор. 10).

## ВИСНОВКИ

В ході виконання лабораторної роботи 10 було вивчено 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, а також відпрацьовані навички написання коду.

## ДОДАТОК А

### Лістинг коду програми

```
#include <iostream>

using namespace std;

struct TTime {
    int hrs, min, sec;
};

struct VVariables {
    int A, B;
};

struct NNums {
    int A, B, C;
};

void param79();
void func(TTime& T);
void AddSec(TTime& T);

void begin22();
void Reverse(VVariables& V);

void boolean12();
void is(NNums& n);
bool isPositive(int A, int B, int C);

int main() {

    int choice = 0;

    while (choice != 4) {

        cout << "Choose task: "
              << "\n1.Param79"
              << "\n2.Begin22"
              << "\n3.Boolean14"
              << "\n4.Exit" << endl;

        cin >> choice;

        switch (choice) {
        case 1: {
            param79();          //Task param79
            break;
        }
        case 2: {
```

```

        begin22();          //Task begin22
        break;
    }
    case 3: {
        boolean12();        //Task boolean12
        break;
    }
    case 4: {
        cout << "Program is end!";
        break;
    }
    default: {
        cout << "Wrong one, choose again!\n";
    }
}

}

//Start param79

void param79() {
    TTime t1, t2, t3, t4, t5;
    func(t1);
    func(t2);
    func(t3);
    func(t4);
    func(t5);
}

void func(TTime& T) {
    cout << "Enter hours: ";
    cin >> T.hrs;

    cout << "Enter minutes: ";
    cin >> T.min;

    cout << "Enter a seconds: ";
    cin >> T.sec;

    if (T.hrs < 24 && T.min < 60 && T.sec < 60) {
        AddSec(T);
        cout << "Edited hours:\n ";
        cout << T.hrs << ":" << T.min << ":" << T.sec << "\n";
    }
    else if (T.hrs > 23) {
        cout << "\nHours > 24!";
    }
    else if (T.min > 59) {
        cout << "\nMinutes > 59!";
    }
}

```

```

    }
    else if (T.hrs > 59) {
        cout << "\nSeconds > 59!";
    }
}

void AddSec(TTime& T) {
    int N;

    cout << "How much secs you want to add: ";
    cin >> N;
    int temp = 0;

    do {
        N -= 60;
        T.min++;
    } while (N > 60);
    T.sec += N;
    if (T.sec >= 60) {
        temp = T.sec / 60;
        T.sec %= 60;
        T.min += temp;
    }
    if (T.min >= 60) {
        temp = T.min / 60;
        T.min %= 60;
        T.hrs += temp;
    }
    if (T.hrs >= 24) {
        temp = T.hrs / 24;
        T.hrs %= 24;
    }
    do {
        T.hrs;
        T.min - 60;
    } while (T.min > 60);
}

//End Param79

//Start begin22

void begin22() {
    VVariables V;
    Reverse(V);
}

void Reverse(VVariables& V) {
    cout << "Enter A: ";
    cin >> V.A;
    cout << "Enter B: ";

```



```

    cin >> V.B;

    swap(V.A, V.B);

    cout << "A = " << V.A << endl;
    cout << "B = " << V.B << endl;

}

//End begin22

//Start boolean12
void boolean12() {
    NNums n;
    is(n);
}

void is(NNums& N) {
    cout << "Enter A: ";
    cin >> N.A;
    cout << "Enter B: ";
    cin >> N.B;
    cout << "Enter C: ";
    cin >> N.C;

    cout << "Each number from these are positive = " << boolalpha <<
isPositive(N.A, N.B, N.C) << endl;
}

bool isPositive(int A, int B, int C) {
    return (A > 0 && B > 0 && C > 0);
}

//End boolean12

```

ДОДАТОК Б  
Скрін-шот вікна виконання програми

```
Choose task:
1.Param79
2.Begin22
3.Boolean14
4.Exit
1
Enter hours: 21
Enter minutes: 22
Enter a seconds: 14
How much secs you want to add: 420
Edited hours:
 21:29:14
Enter hours: 13
Enter minutes: 45
Enter a seconds: 12
How much secs you want to add: 337
Edited hours:
 13:50:49
Enter hours: 7
Enter minutes: 40
Enter a seconds: 30
How much secs you want to add: 920
Edited hours:
 7:55:50
Enter hours: 11
Enter minutes: 50
Enter a seconds: 21
How much secs you want to add: 1400
Edited hours:
 12:13:41
Enter hours: 23
Enter minutes: 54
Enter a seconds: 45
How much secs you want to add: 600
Edited hours:
 0:4:45
Choose task:
1.Param79
2.Begin22
3.Boolean14
4.Exit
2
Enter A: 4
Enter B: 8
A = 8
B = 4
Choose task:
1.Param79
2.Begin22
3.Boolean14
4.Exit
3
Enter A: 1
Enter B: 2
Enter C: 3
Each number from these are positive = true
Choose task:
1.Param79
2.Begin22
3.Boolean14
4.Exit
```

Рисунок Б - Екран виконання програми для вирішення завдання 1,2