МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

Інститут прикладної математики та фундаментальних наук

Кафедра прикладної математики

**ЗВІТ**

про виконання лабораторних робіт

з дисципліни

***«*Алгоритми та програмування, частина 2*»***

Виконала:  
студентка  
групи ПМ-11  
Вітюк Аліна  
Прийняв:   
Гладун В.Р.

**Лабораторна робота №5**

**Тема:** структури та дії над ними

**Мета:** набуття практичних навичок у конструюванні та використанні структур

**Хід роботи:**

1. Постановку задачі: описати структурну змінну «Дата» з полями «Число», «Місяць» та «Рік». Написати та протестувати функцію, яка:

А) Обчислює інтервал (в днях), що характеризує часову віддаль між обома заданими датами;   
Б) Аналізуючи порядковий номер дня в році, визначає число і місяць року, які відповідають заданому дню;  
В) Аналізуючи введену дату, визначає дату на N днів вперед.

1. Обґрунтування алгоритму:

У пункті А потрібно обчислити інтервал (в днях) між датами, що задав користувач. Для цього потрібно обчислити скільки секунд пройшло від 01.01.1970 до заданих дат. Далі знайти різницю між отриманими значеннями й поділила її на кількість секунд в одному дні. Отримане число й буде інтервалом між датами.   
У пункті Б для визначення дати за порядковим номером дня у році використала масив з кількістю днів у кожному місяці. Далі потрібно робити перевірку, якщо задане число більше за кількість днів у місяці, збільшувався порядковий номер місяця. І так поки не буде встановлена дата, що відповідає заданому користувачу дню.   
У пункті В для визначення дати на N днів вперед потрібно до введеного користувачем значення дню додавати N (для зручності позначу значення “День” + N, як D). Далі слід перевірити чи отримане значення D не перевищує допустиму кількість днів у введеному користувачем місяці. Якщо так, то збільшується лічильник для місяців. Від значення D віднімається кількість днів, що відповідає даному місяцю, даний алгоритм повторюється поки значення D не буде дорівнювати 0. Якщо значення місяцю перевищить 12, збільшується лічильник для року і починається з початку відлік місяців.

1. Ідентифікатори:  
   Day – день;  
   Month – місяць;  
   Year – рік;  
   mday – кількість днів в кожному місяці;  
   secondsInDay – кількість секунд в одному дні;  
   Date\_1, Date\_2 – дати між якими шукаємо інтервал у пункті А;  
   diff – шуканий інтервал;  
   d – порядковий номер дня, за яким визначаємо дату у пункті Б;  
   resultDate – дата, визначена за порядковим номером дня в році;  
   N – крок з яким визначаємо нову дату у пункті В;  
   Date\_3 – введена користувачем дата від якої визначає дату на N днів вперед;  
   resultDate2 – дата з кроком N.
2. Текст документованої програми:

#include <iostream>

#include <ctime>

#include<Windows.h>

using namespace std;

struct Date {

int Day; //день

int Month; //місяць

int Year; //рік

};

const int secondsInDay = 60 \* 60 \* 24; //кількість секунд в дні

long dateInterval(const Date& startDate, const Date& endDate);   
//Обчислює інтервал(в днях), що характеризує часову віддаль між обома заданими датами;

Date dateFromDayOfYear(int day);   
//Аналізуючи порядковий номер дня в році, визначає число і місяць року, які відповідають заданому дню;

int daysInMonth(int mday);   
// Кілкість днів у місяці

Date leapAhead(const Date& sourceDate, int daysAhead);   
//Аналізуючи введену дату, визначає дату на N днів вперед.

int EnterNumber(const char\*); // введення числа

Date EnterDate(const char\*); // введення дати

bool IncorrectDate(Date); // перевірка коректності дати

int main()

{

SetConsoleOutputCP(1251); /\* зміна кодування для використання кирилиці при вводі та виводі \*/

SetConsoleCP(1251);

// пункт A

cout << "\nПОШУК ІНТЕРВАЛУ МІЖ ДАТАМИ\n";

Date Date\_1;

Date Date\_2;

Date\_1 = EnterDate("першу дату");

Date\_2 = EnterDate("другу дату");

auto diff = dateInterval(Date\_1, Date\_2);

cout << "\nІнтервал між датами:" << diff;

cout << "\n";

// пункт B

cout << "\nВИЗНАЧИТИ ДАТУ ЗА ПРЯДКОВИМ НОМЕРОМ ДНЯ\n";

int d = 0;

do {

d = EnterNumber("порядковий номер дня в 2021 році:");

} while (d > 365);

Date resultDate = dateFromDayOfYear(d);

cout << "\nДаному дню відповідає дата:" << resultDate.Day << "." << resultDate.Month << ".2021";

cout << "\n";

// пункт C

cout << "\nВИЗНАЧИТИ ДАТУ НА N ДНІВ ВПЕРЕД\n";

Date Date\_3;

Date\_3 = EnterDate("дату");

int N;

N = EnterNumber("крок N : ");

Date resultDate2 = leapAhead(Date\_3, N);

cout << "\nДата з кроком N: " << resultDate2.Day << "." << resultDate2.Month << "." << resultDate2.Year;

return 0;

}

// введення дати

Date EnterDate(const char\* ms) {

Date a;

cout << "\n\tВводьте " << ms << " :\n";

do {

a.Day = EnterNumber(" день ");

a.Month = EnterNumber(" мiсяць ");

a.Year = EnterNumber("рiк");

} while (IncorrectDate(a)); // перевірка коректності дати

return a;

}

// ітервал між датами

long dateInterval(const Date& Date\_1, const Date& Date\_2) {

struct tm tm\_1;

tm\_1.tm\_min = 0;

tm\_1.tm\_hour = 0;

tm\_1.tm\_sec = 0;

tm\_1.tm\_year = Date\_1.Year - 1900;

tm\_1.tm\_mon = Date\_1.Month - 1;

tm\_1.tm\_mday = Date\_1.Day;

time\_t t\_1 = mktime(&tm\_1);

struct tm tm\_2;

tm\_2.tm\_min = 0;

tm\_2.tm\_hour = 0;

tm\_2.tm\_sec = 0;

tm\_2.tm\_year = Date\_2.Year - 1900;

tm\_2.tm\_mon = Date\_2.Month - 1;

tm\_2.tm\_mday = Date\_2.Day;

time\_t t\_2 = mktime(&tm\_2);

long differenceInSeconds = t\_2 - t\_1;

differenceInSeconds / secondsInDay;

return differenceInSeconds / secondsInDay;

}

// пошук дати за днем в році

Date dateFromDayOfYear(int d) {

Date resDate;

int dayOfTheYear = d;

int month = 1;

int days = 0;

while (dayOfTheYear > 0) {

days += daysInMonth(month);

dayOfTheYear = d - days;

month++;

}

resDate.Day = d - days + daysInMonth(month - 1);

resDate.Month = month - 1;

resDate.Year = 0;

return resDate;

}

// днів в місяці

int daysInMonth(int mday) {

int mDays[] = { 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };

return mDays[mday % 12];

}

// дата з кроком N

Date leapAhead(const Date& Date\_3, int N) {

Date date = Date\_3;

date.Day += N;

int month = date.Month;

while (date.Day > daysInMonth(month - 1)) {

date.Day -= daysInMonth(month + 1);

month++;

}

date.Month = month;

while (date.Month > 12) {

date.Month -= 12;

date.Year++;

}

return date;

}

// перевірка коректності введеної дати

bool IncorrectDate(Date a)

{

if (a.Month < 1 || a.Month > 12 || a.Day < 1 || a.Day > 31)

{

cout << "\nНекоректна дата\n";

return true;

}

bool rez = false; /\* true - якщо дата некоректна, false - якщо дата коректна \*/

switch (a.Month)

{

case 4: case 6: case 9: case 11: if (a.Day > 30) rez = true; break; /\* місяці, які мають по 30 днів \*/

case 2: /\* місяць лютий \*/

if (a.Day > 29)

rez = true;

else

rez = (a.Day == 29) && (a.Year % 4 || (a.Year % 100 == 0 && a.Year % 400)); break; /\* не високосний рік \*/

}

if (rez == true)

{

cout << "\nНекоректна дата\n";

}

return rez;

}

// введення числа

int EnterNumber(const char\* ms) {

cout << "\nВводьте " << ms << " = ";

char rez[15];

bool flag = false;

do {

flag = false;

gets\_s(rez);

for (int i = 0; i < strlen(rez); i++) {

if (!isdigit(rez[i])) { flag = true; }

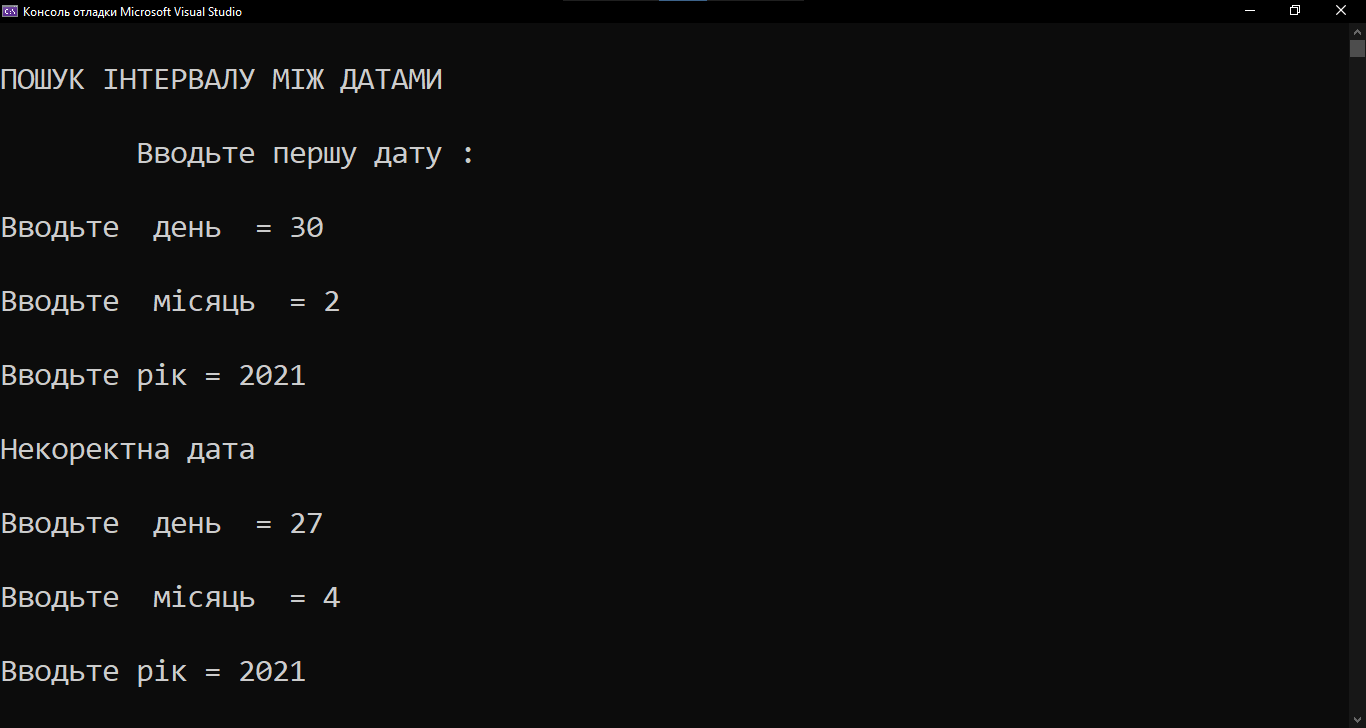
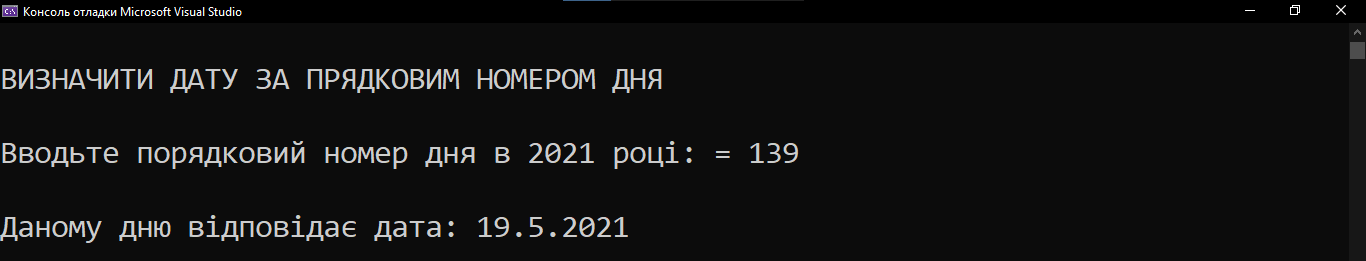
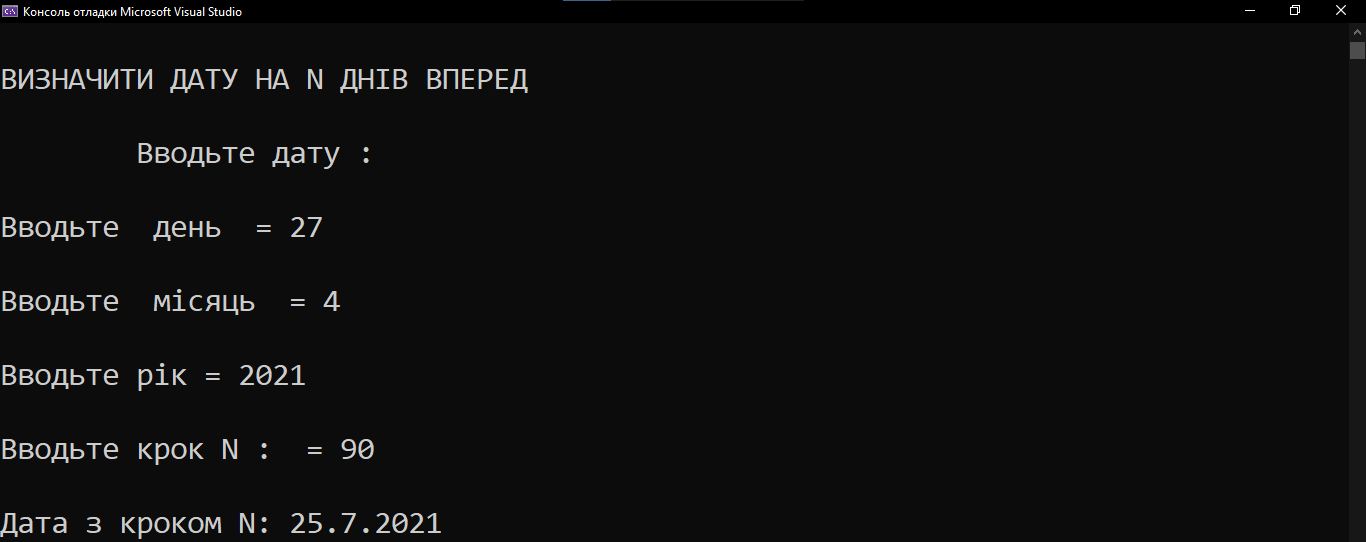
}

if (flag == true) cout << "\nВведіть коректне число: ";

} while (flag == true);

return atoi(rez);

}

1. Результат виконання програми:  
     
     
   
2. Програма призначена для:

А) Обчислення інтервалу (в днях), що характеризує часову віддаль між двома заданими датами;

Після запуску коду програми на екран, користувачу необхідно ввести з клавіатури дві бажані дати (при введені некоректної дати на екран виводиться відповідне повідомлення і користувачу доведеться ввести іншу дату). Після успішного введення даних, на екран виведеться число, що характеризує часову віддаль між заданими датами.

Б) Аналізу порядкового номеру дня в році, визначення числа і місяць року, які відповідають заданому дню;

Користувачу слід ввести порядковий номер дня в році, для якого буде визначено число і місяць. Введене число не може перевищувати кількість днів в році. Після введення, на екран виведеться дата, що відповідаю заданому користувачем дню.

В) Аналізу введеної дати, визначення дати на N днів вперед.

Користувачу необхідно ввести дату і крок N. Після успішного введення даних, на екран виведення нова дата.

1. Можливі модифікації програми: додати меню, передбачити ввід року користувачем у пункті Б.

**Висновок:** виконання лабораторної роботи №5 допомогло засвоїти практичні навички у роботі зі структурами. Також я навчилася виконувати різні операції з структурою дати.