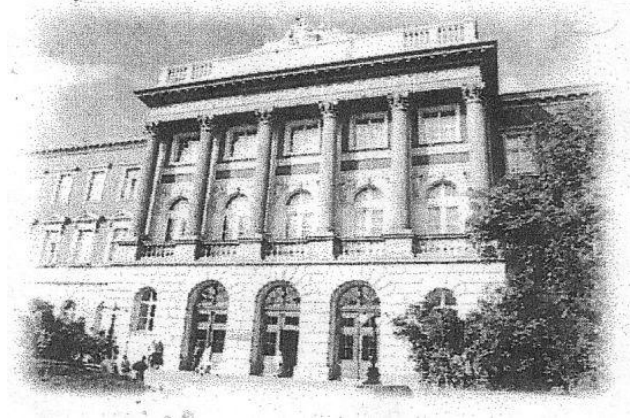


Міністерство освіти і науки України
Національний університет «Львівська політехніка»

L I T T E R I S E T A R T I B V S



**Програмування на С алгоритмів розгалуженої структури.
Програма для розв'язування алгебраїчних нерівностей.**

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторної роботи № 3

з дисципліни «Основи інформатики та програмування»
для студентів спеціальності 105 – "Прикладна фізика та наноматеріали"
для першого (бакалаврського) рівня освіти.

Затверджено
на засіданні кафедри
обчислювальної математики
та програмування
Протокол № 10 від 07.06.2018р.

Львів – 2018

Програмування на С алгоритмів розгалуженої структури. Програма для розв'язування алгебраїчних нерівностей. Методичні вказівки до виконання лабораторної роботи № 3 для студентів спеціальності 105 – "Прикладна фізика та наноматеріали" для першого (бакалаврського) рівня освіти / Укл.: Гнатів Л.Б., Ментинський С.М., Пелех Я.М., Угрин С.З., 2018. - 12 с.

Реєстр. № 8184 від 19.06.2018

Укладачі:

Гнатів Л.Б., к. ф-м. н., доцент,
Ментинський С.М., ст. викл,
Пелех Я.М., к. ф-м. н., доцент,
Угрин С.З., асистент.

Відповідальний за випуск: Угрин С.З.

Рецензенти:

Будз І.С., к.ф-м.н., доцент кафедри ОМП,
Філь Б.М., к.ф-м.н., доцент кафедри ОМП.

Передмова

У методичних вказівках розглянуто теоретичні та практичні аспекти побудови розгалужених алгоритмів для розв'язування нескладних практичних задач, та їх реалізацію за допомогою мов програмування С та С++.

Методичні вказівки містять коротку довідкову інформацію, варіанти індивідуальних завдань та приклад розв'язування завдання одного з варіантів з детальним описом.

Методичні вказівки призначені для студентів спеціальності 105 – "Прикладна фізика та наноматеріали" для першого (бакалаврського) рівня освіти і укладені відповідно до робочої навчальної програми з дисципліни «Основи інформатики та програмування».

ЛАБОРАТОРНА РОБОТА № 3.

Програмування на С алгоритмів розгалуженої структури. Програма для розв'язування алгебраїчних нерівностей.

Мета роботи: Ознайомлення з алгоритмічною конструкцією розгалуження в мовах С/С++ та її використанням в програмах. Отримання практичних навичок програмування алгоритмів розгалуженої структури.

Короткі теоретичні відомості.

Умовна інструкція С/С++.

Умовна інструкція має форму:

```
if (логічний вираз) інструкція;
```

Якщо логічний вираз істинний (true) то буде виконана інструкція за умовою, інакше вона не буде виконана. Якщо необхідно виконати декілька інструкцій, то їх розміщують у блоці:

```
if (умова) {  
    інструкція 1;  
    ...  
    інструкція n;  
}
```

Якщо ж необхідно здійснити певну дію в разі хибності логічного виразу, то застосовують умовну інструкцію наступного виду:

```
if (умова) інструкція1;  
else інструкція2;
```

Тернарний оператор

У випадку, коли умовна конструкція використовується безпосереднього для обчислення значення деякого виразу, зручніше користуватися однією із спеціальних форм оператора присвоювання, що веде своє походження від умовного арифметичного виразу, що вперше з'явився в мові АЛГОЛ-60:

```
v = (a > b) ? e1 : e2;
```

Виконується такий оператор наступним чином. Перевіряється умова, записана в круглих дужках і, якщо вона виконується, то змінній *v* присвоюється значення виразу *e1*. В іншому випадку у змінну *v* заноситься значення виразу *e2*. Така конструкція у програмуванні носить назву тернарного оператора.

Логічні операції та вирази

Для побудови простих умов в розгалуженнях чи циклах використовують оператори відношення (порівняння):

== дорівнює; != не рівне; > більше < менше;
>= більше або рівне <= менше або рівне

Результатом операції порівняння в C++ є результат типу bool із значеннями true або false. У мові C логічні змінні відсутні, а замість цього діє домовленість про те, що брехня еквівалентна нульовому значенню числового виразу, а істина - не нульовому. Числові дані в C++ також можна використовувати в якості умови оператора if, - у цьому випадку відбувається перетворення(приведення типу) до bool за правилом: 0=false, число відмінне від 0, перетворюється на true.

. Оператор вибору (перемикач)

Оператор вибору розширює можливості умовного оператора if, за його допомогою можна організувати розгалуження на багато віток. Наприклад, найпростіший калькулятор можна було б змодельювати таким чином:

```
switch (/*змінна або вираз*/)
{
    case /*константний вираз1*/: /*група операторов*/;
        break;
    case /* константний вираз 2*/: /*група операторов*/;
        break;
    // . . .
    default: /*група операторов*/;
}
```

Після службового слова switch (від англ. - Перемикач) в круглих дужках записується перемикаючий вираз, що може бути цілочисельним або символьним. Тіло перемикача завжди є складеним оператором і береться у фігурні дужки. За службовим словом case (від англ. - Випадок) записується константа, з якою порівнюється значення перемикаючого виразу. Якщо збіг знайдено, то управління передається оператору, відокремленому від константи двокрапкою. На його місці може виявитися або один оператор, або ланцюжок операторів. Якщо значення не збігається із зазначеною константою, то з тіла перемикача вибирається case, і т.д.

Останнім рядком перемикача може бути рядок, що починається з службового слова default. На нього управління потрапляє, якщо значення не збігається з жодною з передбачених констант.

Відсутність оператора break в якості завершальної дії якої-небудь групи приведе до того, що почнуть виконуватися оператори, що належать наступній групі (частина наступного рядка "case c:" просто ігнорується).

Завдання 1. *Написати програму для розв'язання задачі згідно варіанту. Всі дані для роботи програми вводити з консолі. Якщо тип даних в умові не вказано явно, то вважати дані дійсними числами типу double.*

1. Дано три цілих числа. Знайти кількість додатних чисел в цьому наборі.
2. Дано три дійсних числа. Знайти кількість від'ємних чисел в цьому наборі.
3. Дано дві змінні дійсного типу: А, В. Перерозподілити значення даних змінних так, щоб А виявилось меншим із заданих значень, а В - більше. Вивести нові значення змінних А і В.
4. Дано дві змінні цілого типу: А і В. Якщо їхні значення не рівні, то присвоїти кожній змінній суму цих значень, а якщо рівні, то присвоїти змінним нульові значення. Вивести нові значення змінних.
5. Дано три числа. Знайти найменше з них.
6. Дано три числа. Знайти середнє з них (тобто число, розташоване між найменшим і найбільшим).
7. Дано три числа. Вивести спочатку найменше, а потім найбільше з даних чисел.
8. Дано три числа. Знайти суму двох більших з них.
9. Дано координати точки, що не лежить на координатних осях ОХ та ОУ. Визначити номер координатної чверті, в якій знаходиться дана точка.
10. Дано цілочисельні координати трьох вершин прямокутника, сторони якого паралельні координатним осям. Знайти координати його четвертої вершини.
11. Тіло масою m з об'ємом V занурюють в рідину густиною ρ . Дослідити чи тіло плаватиме в рідині і вивести відповідне повідомлення.
12. Для введених двох чисел визначити найбільшу величину серед їх суми, різниці та добутку.
13. Для введених двох чисел визначити найбільшу величину серед їх суми, добутку та частки.
14. Дано три числа. Знайти суму двох менших з них.
15. Дано три числа. Знайти різницю між найбільшим і найменшим з них.

16. Задано натуральне число з діапазону 1 – 9999. Вивести його опис у вигляді: «одноцифрове число», «двоцифрове число» і т. д.

Приклад виконання завдання 1.

Завдання: 16. Задано натуральне число з діапазону 1 – 9999. Вивести його опис у вигляді: «одноцифрове число», «двоцифрове число» і т. д.

Можна побудувати декілька алгоритмів розв’язання поставленої задачі. Зупинимося на варіанті, в якому спочатку перевіримо чи введене число відповідає умові задачі (лежить в діапазоні від 1 до 9999). Якщо введене число дійсно належить вказаному діапазону, то подальшу перевірку можна здійснювати повною формою розгалуження за схемою: числа менші за 10 – одноцифрові, інші, але менші за 100, – двоцифрові, інші, але менші за 1000 – трицифрові, всі решта – чотирицифрові.

Код програми.

```
#include<iostream>
using namespace std;
void main()
{
    setlocale(LC_ALL, "ukrainian");//встановлює правильний вивід
                                   //українських літер в потоці cout;
    int n;
    cout<<"Введіть натуральне число від 1 до 9999\n";
    cin>>n;
    if (n > 0 && n<10000) {
        if(n < 10) cout<<"Одноцифрове число\n";
        else {
            if (n < 100) cout<<"Двоцифрове число\n";
            else {
                if (n < 1000) cout<<"Трицифрове число\n";
                else cout<<"Чотирицифрове число\n";
            }
        }
    }
    else cout<<"Введене число лежить за межами вказаного діапазону\n";
}
```

Завдання 2. Скласти програму для розв'язування алгебраїчної нерівності (системи нерівностей) згідно варіанту. Програма повинна для заданих значень числових коефіцієнтів a , b та c визначати розв'язок у вигляді проміжків на числовій осі. Виконати програму для різних значень коефіцієнтів a , b і c так, щоб продемонструвати її роботу за усіма можливими вітками розгалуження.

Варіанти завдань

№ вар.	Завдання	№ вар.	Завдання
1	$\frac{a}{x^2 + bx + c} \leq 0$	9	$\frac{a}{x^2 + bx + c} > 0$
2	$ax^2 - bx > c$	10	$ax^2 - bx \leq c$
3	$\frac{x^2}{ax + b} < c$	11	$\frac{x^2}{ax + b} \geq c$
4	$\frac{x - a}{x^2 + bx + c} \geq 0$	12	$\frac{x - a}{x^2 + bx + c} < 0$
5	$\frac{x^2 - ax + b}{x + c} \leq 0$	13	$\frac{x^2 - ax + b}{x + c} > 0$
6	$(x - a)(x^2 + bx + c) < 0$	14	$(x + a)(x^2 - bx + c) > 0$
7	$\begin{cases} x - a < 0, \\ x^2 + bx + c > 0 \end{cases}$	15	$\begin{cases} x + a > 0, \\ x^2 + bx + c \leq 0 \end{cases}$
8	$\begin{cases} x - a \geq 0, \\ x^2 + bx + c < 0 \end{cases}$	16	$\begin{cases} x - a \geq 0, \\ x^2 + bx + c > 0 \end{cases}$

Приклад виконання завдання 2. (варіант № 16)

$$\begin{cases} x - a \geq 0, \\ x^2 + bx + c > 0 \end{cases}$$

Результатом виконання роботи має бути програма, яка отримавши від користувача конкретні числові значення коефіцієнтів a , b , c , видавала б розв'язок відносно x нерівності у вигляді проміжків на числовій осі. Головною запорукою правильної роботи програми є ґрунтовний аналіз можливих варіантів процесу розв'язування нерівності, залежно від значень коефіцієнтів a , b та c .

Перша нерівність має розв'язок $x \geq a$, тобто проміжок $[a; +\infty)$. Розв'язок системи залежить від того яким є розв'язок другої нерівності, та як відносно цього розв'язку розташоване на числовій осі число a . Тобто програма повинна обчислити значення дискримінанта D квадратного тричлена в лівій частині другої нерівності та встановити наявність-відсутність його коренів.

Якщо дискримінант від'ємний, то квадратний тричлен не має коренів, а оскільки коефіцієнт при x^2 додатний, то друга нерівність має розв'язком усю множину дійсних чисел \mathbb{R} . Отже розв'язком системи буде проміжок $[a; +\infty)$.

При $D=0$ квадратний тричлен в другій нерівності має коренем число $x_0 = \frac{-b}{2}$, а сама нерівність – розв'язок $(-\infty; x_0) \cup (x_0; +\infty)$. Тоді розв'язок системи залежить від взаємного розташування на числовій прямій чисел a та x_0 :

- при $x_0 < a$ розв'язком системи залишиться проміжок $[a; +\infty)$;
- при $x_0 = a$ число a не є розв'язком системи і результатом буде $x \in (a; +\infty)$ (те ж що і $x \in (x_0; +\infty)$);
- при $x_0 > a$ число x_0 слід виключити з проміжка $[a; +\infty)$, отримуємо результат $x \in [a; x_0) \cup (x_0; +\infty)$

При $D > 0$ знайдемо корені $x_1 = \frac{-b - \sqrt{D}}{2}$ та $x_2 = \frac{-b + \sqrt{D}}{2}$ (очевидно, що $x_1 < x_2$)

квадратного тричлена в другій нерівності. Нерівність матиме розв'язок $(-\infty; x_1) \cup (x_2; +\infty)$ а для розв'язку системи матимемо такі випадки:

- при $a < x_1$ розв'язок системи $x \in [a; x_1) \cup (x_2; +\infty)$;
- при $x_1 \leq a \leq x_2$ розв'язок системи $x \in (x_2; +\infty)$;
- при $a > x_2$ розв'язком системи залишиться проміжок $[a; +\infty)$.

Враховуючи проведений аналіз, код функції для розв'язування нерівності можна оформити наступним чином:

```
#include<iostream>
#include<math.h>
using namespace std;

void main()
{
    double a,b,c,d;
    setlocale(LC_ALL, "ukrainian");//встановлює правильний вивід
    //українських літер в потоці cout;
    cout<<"Введіть коефіцієнти a,b,c нерівності\n";
    cin>> a>>b>>c;
    d = b*b-4*c;
    if (d<0)
        cout<< "["<<a<<" ;+inf)\n";
    else{
        if (d==0){
            double x0 = -b/2;
            if (a < x0) cout<<"x ∈ ["<<a<<
                "<< x0 << ")U("<<x0<<" ;+inf)\n";
            else {
                if(a == x0) cout<<"x ∈ ("<<a<<" ;+inf)\n";
                else cout<<"x ∈ ["<<a<<" ;+inf)\n";
            }
        }
    }
}
```

```

else{
    double x1 = (-b-sqrt(d))/2;
    double x2 = (-b+sqrt(d))/2;
    if (a < x1) cout<<"x ∈ ["<<a<<"
        << x1 <<")U("<<x2<<"+inf)\n";
    else{
        if (a <= x2) cout<<"x ∈ ("<<x2<<"+inf)\n";
        else cout<<"x ∈ ["<<a<<"+inf)\n";
    }
}
}
}
}

```

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Який синтаксис умовної конструкції в C/C++?
2. Який результат операцій відношення в мові C++?
3. Яке призначення оператора *else*?
4. В чому полягає відмінність між повною та короткою формою розгалуження?
5. Що таке тернарний оператор (операція)?
6. Які дії виконує оператор switch ?
7. Яке призначення блоку default в операторі switch?

Рекомендована література:

1. Програмування на С(С++). Парадигма процедурного програмування. Навчальний посібник. /Гнатів Б.В., Гнатів Л.Б. . – Львів: Видавництво «Растр-7», 2017. – 264 с.
2. Основи програмування на С/С++. Конспект лекцій з курсу «Основи інформатики і програмування, частина 2» спеціальності 105 – «Прикладна фізика та наноматеріали» для першого (бакалаврського) рівня освіти/ Укл.: Ментинський С.М., 2016. – 140 с
3. П. Каленюк, І. Ключник, І. Кравець, Л. Гнатів, Л.Демків, І. Подун Основи програмування в середовищі С. Лекції та завдання до лабораторних робіт. Львів 2007 р., 98 с.
4. Петрович Р. Й., Тумашова О. В. “Основи програмування мовою Сі.”. Навчальний посібник. Видавництво НУ”ЛП”, Львів, 2005р. 100с.
5. Програмування мовою С: навчальний посібник для вузів / Зореслава Ярославівна Шпак . — Львів: Оріяна-Нова, 2006 . — 431 с.
6. Кравець П.О. Об’єктно-орієнтоване програмування: навч.посібн. / П.О.Кравець.— Львів: Вид-во Львівської політехніки, 2012.— 624 с.
7. Крєневич, А.П. С у задачах і прикладах : навчальний посібник із дисципліни "Інформатика та програмування" / А.П. Крєневич, О.В. Обвінцев. – К. : Видавничо-поліграфічний центр "Київський університет", 2011. – 208 с.
8. Шилдт, Герберт. Полный справочник по С++, 4-е издание. . Пер. с англ. — М. : Издательский дом “Вильямс”, 2006. — 800 с. : ил.
9. Стивен Пратта Язык программирования С. Лекции и упражнения - К.: Диасофт, 2002.

НАВЧАЛЬНЕ ВИДАННЯ

Програмування на C алгоритмів розгалуженої структури. Програма для розв'язування алгебраїчних нерівностей.

МЕТОДИЧНІ ВКАЗІВКИ до виконання лабораторної роботи №3 з дисципліни «Основи інформатики та програмування»

для студентів спеціальності 105 – "Прикладна фізика та наноматеріали" для першого (бакалаврського) рівня освіти.

Укладачі:

Гнатів Л.Б., к. ф-м. н., доцент,
Ментинський С.М., ст. викл,
Пелех Я.М., к. ф-м. н., доцент,
Угрин С.З., асистент.

Комп'ютерне складання: Ментинський С.М., ст. викл.