

Лабораторна робота № 3

Тема: Застосування алгоритму бінарного пошуку.

Мета роботи: набуття практичних навичок застосування алгоритму бінарного пошуку.

Хід роботи

Завдання для виконання

- Виконати Завдання 1 та перевірити роботу програми для отримання розв'язку нелінійного чи трансцендентного рівняння (підібрати самостійно). Результати занести у звіт.

Завдання 1. Реалізуйте алгоритм дійсного бінарного пошуку із завершенням роботи при безпосередньому сусістві двох дійсних чисел.

Лістинг 3.1 – Код програми 1

```
//Чавурська Єлизавета, ЛРЗ, Варіант 20, АiСД
#include <iostream>
#include <cmath>
using namespace std;
double f(double x) {
    return x * x * x - 4 * x + 1;
}
// Реалізація дійсного бінарного пошуку
double Func1(double left, double right, double eps) {
    double mid;
    while (fabs(right - left) > eps) {
        mid = (left + right) / 2;
        // Якщо функція змінює знак у середині відрізка – звужуємо інтервал
        if (f(left) * f(mid) <= 0)
            right = mid;
        else
            left = mid;
    }
    // Повертаємо середину останнього інтервалу
    return (left + right) / 2;
}
int main() {
    setlocale(LC_CTYPE, "ukr");
    double a, b, eps;
    cout << "Введіть початок діапазону a: ";
    cin >> a;
    cout << "Введіть кінець діапазону b: ";
    cin >> b;
    cout << "Введіть точність (наприклад 0.0001): ";
```

```

    cin >> eps;
    double result = Func1(a, b, eps);
    cout << "\nКорінь рівняння f(x) = 0: x = " << result << endl;
    cout << "Перевірка: f(x) = " << f(result) << endl;
    cout << "Чавурська Єлизавета, ЛРЗ, Варіант 20, Завдання 1" << endl;
    return 0;
}

```

```

Введ?ть початок д?апазону a: 1
Введ?ть к?нець д?апазону b: 6
Введ?ть точн?сть (наприклад 0.0001): 2

Кор?нь р?вняння f(x) = 0: x = 1.625
Перев?рка: f(x) = -1.20898
Чавурська Єлизавета, ЛРЗ, Вар?ант 20, ЗАвдання 1

```

Рисунок 3.1 – Вікно результатів програми 1

Лістинг 3.2 – Код програми 2 (перевірка роботи програми для отримання розв'язку нелінійного чи трансцендентного рівняння)

```

//Чавурська Єлизавета, ЛРЗ, Варіант 20, AiСД
#include <iostream>
#include <cmath>
using namespace std;
// Нелінійна функція
double f1(double x) {
    return x * x * x - 4 * x + 1; // приклад нелінійного рівняння
}
// Трансцендентна функція
double f2(double x) {
    return sin(x) - 0.5; // приклад трансцендентного рівняння
}
// Реальний бінарний пошук
double Func1(double left, double right, double eps, double(*f)(double)) {
    double mid;
    while (fabs(right - left) > eps) {
        mid = (left + right) / 2;
        if (f(left) * f(mid) <= 0)
            right = mid;
        else
            left = mid;
    }
    return (left + right) / 2;
}
int main() {

```

```

setlocale(LC_CTYPE, "ukr");
double a, b, eps;
int choice;
cout << "Оберіть тип рівняння:\n";
cout << "1 - Нелінійне ( $x^3 - 4x + 1 = 0$ )\n";
cout << "2 - Трансцендентне ( $\sin(x) - 0.5 = 0$ )\n";
cout << "Ваш вибір: ";
cin >> choice;
cout << "Введіть початок діапазону a: ";
cin >> a;
cout << "Введіть кінець діапазону b: ";
cin >> b;
cout << "Введіть точність: ";
cin >> eps;
double result;
if (choice == 1) {
    result = Func1(a, b, eps, f1);
    cout << "\nРозв'язок нелінійного рівняння  $x^3 - 4x + 1 = 0$ : \n";
    cout << "x ~ " << result << "\n";
    cout << "Перевірка: f(x) = " << f1(result) << endl;
}
else {
    result = Func1(a, b, eps, f2);
    cout << "\nРозв'язок трансцендентного рівняння  $\sin(x) - 0.5 = 0$ : \n";
    cout << "x ~ " << result << "\n";
    cout << "Перевірка: f(x) = " << f2(result) << endl;
}
cout << "Чавурська Єлизавета, ЛРЗ, Варіант 20, Завдання 1" << endl;
return 0;
}

```

Обер?ть тип р?вняння:
1 – Нел?н?йне ($x^3 - 4x + 1 = 0$)
2 – Трансцендентне ($\sin(x) - 0.5 = 0$)
Ваш виб?р: 2
Введ?ть початок д?апазону a: 1
Введ?ть к?нець д?апазону b: 8
Введ?ть точн?сть: 4

Розв'язок трансцендентного р?вняння $\sin(x) - 0.5 = 0$:
x ? 2.75
Перев?рка: f(x) = -0.118339
Чавурська Єлизавета, ЛРЗ, Вар?ант 20, Завдання 1

Рисунок 3.2 – Вікно результатів програми 2

2. Доповнити програму виконання Завдання 1 підрахунком кількості необхідних ітерацій до завершення роботи. Результати занести у звіт.

Лістинг 3.3 – Код програми 3

```
// Чавурська Єлизавета, ЛРЗ, Варіант 20, AiСД
#include <iostream>
#include <cmath>
using namespace std;

// Нелінійна функція
double f1(double x) {
    return x * x * x - 4 * x + 1; // приклад нелінійного рівняння
}

// Трансцендентна функція
double f2(double x) {
    return sin(x) - 0.5; // приклад трансцендентного рівняння
}

// Реальний бінарний пошук з підрахунком ітерацій
double Func1(double left, double right, double eps, double(*f)(double), int &count) {
    double mid;
    count = 0;
    while (fabs(right - left) > eps) {
        count++;
        mid = (left + right) / 2;
        if (f(left) * f(mid) <= 0)
            right = mid;
        else
            left = mid;
    }
    return (left + right) / 2;
}

int main() {
    setlocale(LC_CTYPE, "ukr");
    double a, b, eps;
    int choice;

    cout << "Оберіть тип рівняння:\n";
    cout << "1 - Нелінійне ( $x^3 - 4x + 1 = 0$ )\n";
    cout << "2 - Трансцендентне ( $\sin(x) - 0.5 = 0$ )\n";
    cout << "Ваш вибір: ";
    cin >> choice;

    cout << "Введіть початок діапазону a: ";
    cin >> a;
    cout << "Введіть кінець діапазону b: ";
    cin >> b;
    cout << "Введіть точність (наприклад 0.0001): ";
}
```

```

    cin >> eps;

    int count = 0;
    double result;

    if (choice == 1) {
        result = Func1(a, b, eps, f1, count);
        cout << "\nРозв'язок нелінійного рівняння  $x^3 - 4x + 1 = 0$ : \n";
        cout << "x ~ " << result << "\n";
        cout << "Перевірка: f(x) = " << f1(result) << endl;
    }
    else {
        result = Func1(a, b, eps, f2, count);
        cout << "\nРозв'язок трансцендентного рівняння  $\sin(x) - 0.5 = 0$ : \n";
        cout << "x~ " << result << "\n";
        cout << "Перевірка: f(x) = " << f2(result) << endl;
    }

    cout << "Кількість ітерацій: " << count << endl;
    cout << "Чавурська Єлизавета, ЛРЗ, Варіант 20, Завдання 1" << endl;
    return 0;
}

```

Оберіть тип рівняння:
1 – Нелінійне ($x^3 - 4x + 1 = 0$)
2 – Трансцендентне ($\sin(x) - 0.5 = 0$)
Ваш вибір: 1
Введіть початок діапазону a: 2
Введіть кінець діапазону b: 23
Введіть точність (наприклад 0.0001): 2.1

Розв'язок нелінійного рівняння $x^3 - 4x + 1 = 0$:
x ~ 22.3438
Перевірка: f(x) = 11066.6
Кількість ітерацій: 4
Чавурська Єлизавета, ЛРЗ, Варіант 20, Завдання 1

Рисунок 3.3 – Вікно результатів програми 3

3. Виконати Завдання 2 та перевірити роботу програми для отримання розв'язку того ж рівняння із точністю обчислення 0,01; 0,001 та 0,00001. Для кожного випадку підрахувати кількість ітерацій. Результати занести у звіт.

Завдання 2. Модифікуйте та реалізуйте алгоритм дійсного бінарного пошуку із завершенням роботи при досягненні точності по аргументу.

Лістинг 3.4 – Код програми 4

```
// Чавурська Єлизавета, ЛРЗ, Варіант 20, АiСД
#include <iostream>
#include <cmath>
using namespace std;
// Нелінійна функція
double f1(double x) {
    return x * x * x - 4 * x + 1; // приклад нелінійного рівняння
}
// Алгоритм дійсного бінарного пошуку з завершенням за точністю по аргументу
double Func1(double left, double right, double eps, double(*f)(double), int &count) {
    double mid;
    count = 0;
    while (fabs(right - left) > eps) {
        count++;
        mid = (left + right) / 2.0;
        if (f(left) * f(mid) <= 0)
            right = mid;
        else
            left = mid;
    }
    return (left + right) / 2.0;
}
int main() {
    setlocale(LC_CTYPE, "ukr");

    double a, b;
    cout << "Розв'язання рівняння x^3 - 4x + 1 = 0\n";
    cout << "Введіть початок діапазону a: ";
    cin >> a;
    cout << "Введіть кінець діапазону b: ";
    cin >> b;
    // Масив
    double epsilons[] = { 0.01, 0.001, 0.00001 };
    int count;
    double result;
    cout << "\nПеревірка роботи алгоритму для різних точностей:\n";
    cout << "-----\n";
    cout << " Точність\t Корінь x\t Кількість ітерацій\n";
    cout << "-----\n";
    for (double eps : epsilons) {
        result = Func1(a, b, eps, f1, count);
        cout << " " << eps << "\t\t" << result << "\t\t" << count << endl;
    }
    cout << "-----\n";
    cout << "Чавурська Єлизавета, ЛРЗ, Варіант 20, Завдання 2" << endl;
    return 0;
}
```

}

Розв'язання рівняння $x^3 - 4x + 1 = 0$

Введіть початок діапазону a: 2

Введіть кінець діапазону b: 21

Перевірка роботи алгоритму для різних точностей:

Точність	Корінь x	Кількість ітерацій
0.01	20.9954	11
0.001	20.9997	15
1e-05	21	21

Чавурська Єлизавета, ЛРЗ, Варіант 20, Завдання 2

Рисунок 3.4 – Вікно результатів програми 4

4. Виконати **Завдання 3** та перевірити роботу програми для отримання розв'язку того ж рівняння із точністю обчислення 0,01; 0,001 та 0,00001. Для кожного випадку підрахувати кількість ітерацій. Результати занести у звіт.

Завдання 3. Модифікуйте та реалізуйте алгоритм дійсного бінарного пошуку із завершенням роботи при досягненні точності по значенню.

Лістинг 3.5 – Код програми 5

```
// Чавурська Єлизавета, ЛРЗ, Варіант 20, АiСД
#include <iostream>
#include <cmath>
#include <iomanip>
using namespace std;
// Нелінійна функція
double f1(double x) {
    return x * x * x - 4 * x + 1; // приклад нелінійного рівняння
}
// Алгоритм дійсного бінарного пошуку з завершенням по значенню функції
double Func1(double left, double right, double eps, double(*f)(double), int &count) {
    double mid;
    count = 0;
    if (f(left) * f(right) > 0) {
        cout << "Метод не може бути застосований: f(a) * f(b) > 0\n";
        return NAN;
    }
}
```

```
while (true) {
    mid = (left + right) / 2.0;
    count++;
    if (fabs(f(mid)) <= eps) // завершення по значенню функції
        break;
    if (f(left) * f(mid) < 0)
        right = mid;
    else
        left = mid;
}
return mid;
}

int main() {
    setlocale(LC_CTYPE, "ukr");
    double a, b;
    cout << "Розв'язання рівняння x^3 - 4x + 1 = 0\n";
    cout << "Введіть початок діапазону a: ";
    cin >> a;
    cout << "Введіть кінець діапазону b: ";
    cin >> b;
    double epsilon[] = { 0.01, 0.001, 0.00001 };
    int count;
    double result;
    cout << "\nПеревірка роботи алгоритму для різних точностей (по значенню
функції):\n";
    cout << "-----\n";
    cout << " Точність\t Корінь x\t Кількість ітерацій\n";
    cout << "-----\n";
    for (double eps : epsilon) {
        result = Func1(a, b, eps, f1, count);
        if (!isnan(result))
            cout << " " << eps << "\t\t" << fixed << setprecision(6) <<
result
            << "\t\t" << count << endl;
    }
    cout << "-----\n";
    cout << "Чавурська Єлизавета, ЛРЗ, Варіант 20, Завдання 3" << endl;
    return 0;
}
```

Розв'язання рівняння $x^3 - 4x + 1 = 0$		
Введіть початок діапазону a: 1		
Введіть кінець діапазону b: 3		
Перевірка роботи алгоритму для різних точностей (по значенню функції):		
Точність	Корінь x	Кількість ітерацій
0.01	1.859375	7
0.001000	1.860840	12
0.000010	1.860806	19
Чавурська Єлизавета, ЛРЗ, Варіант 20, Завдання 3		

Рисунок 3.5 – Вікно результатів програми

5. Реалізувати алгоритм розв'язування задачі Завдання 4 методом бінарного пошуку по відповіді. Протестувати роботу на 4-5 вхідних даних. Результати занести у звіт.

Завдання 4. Реалізуйте алгоритм розв'язування задачі з прикладу 5.1 методом бінарного пошуку по відповіді, використовуючи ідеї, наведені в аналізі.

Лістинг 3.6 – Код програми 6

```
// Чавурська Єлизавета, ЛРЗ, Варіант 20, AiСД
#include <iostream>
#include <cmath>
using namespace std;
bool check(double x) {
    return (x*x >= 10);
}
// Бінарний пошук
double func(double left, double right, double eps) {
    double mid;
    while (right - left > eps) {
        mid = (left + right) / 2.0;
        if (check(mid))
            right = mid; // шукаємо мінімальне x
        else
            left = mid;
    }
    return (left + right) / 2.0;
}

int main() {
    setlocale(LC_CTYPE, "ukr");
    double left, right, eps;
    cout << "Введіть початок інтервалу: ";
    cin >> left;
```

```

cout << "Введіть кінець інтервалу: ";
cin >> right;
cout << "Введіть точність: ";
cin >> eps;

double result = func(left, right, eps);
cout << "Знайдене значення: " << result << endl;

// Тестування на 5 вхідних даних (масив)
double q[5][2] = { {0,10}, {0,20}, {1,5}, {2,10}, {0,100} };
cout << "\nТестування:\n";
for (int i = 0; i < 5; i++) {
    double res = func(q[i][0], q[i][1], eps);
    cout << "Інтервал [" << q[i][0] << ", " << q[i][1] << "] x = " <<
res << endl;
}
cout << "Чавурська Єлизавета, ЛР3, Варіант 20, Завдання 4" << endl;
return 0;
}

```

Введіть початок інтервалу: 1
 Введіть кінець інтервалу: 5
 Введіть точність: 2
 Знайдене значення: 4

Тестування:
 Інтервал [0, 10] x = 3.125
 Інтервал [0, 20] x = 3.125
 Інтервал [1, 5] x = 4
 Інтервал [2, 10] x = 3
 Інтервал [0, 100] x = 3.90625
 Чавурська Єлизавета, ЛР3, Варіант 20, Завдання 4

Рисунок 3.6 – Вікно результатів програми 6

6. Зробити висновок про роботу та записати у звіт.

Висновок: на цій лабораторній роботі я набула практичних навичок застосування алгоритму бінарного пошуку.