МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ
НАВЧАЛЬНО-НАУКОВИЙ КОМПЛЕКС
«ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ»
НАЦІОНАЛЬНОГО ТЕХНІЧНОГО УНІВЕРСИТЕТУ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. ІГОРЯ СІКОРСЬКОГО»
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

Лабораторна робота №1

з курсу «Чисельні методи»

тема: «МЕТОДИ РОЗВ'ЯЗАННЯ НЕЛІНІЙНИХ АЛГЕБРАЇЧНИХ РІВНЯНЬ»

Виконав: студент 2 курсу
групи КА-77
Котів Сергій
Прийняв: Селін О.М.

Київ – 2019р.

Варіант 2

Умова: Знайти всі дійсні корені рівняння:$x^4 - 2x^3 - 9x^2 - 3x - 7 = 0$, попередньо відокремивши їх, а потім із застосуванням чисельних методів уточнити розв'язки.

Допрограмовий етап. Відокремлення коренів рівняння.

Теорема Гюа: Згідно теореми Гюа, рівняння містить комплексні корені оскільки $9 = a_1^1 < a_0 * a_2 = 63$.

Теорема 1:

Оскільки **$a_{0,n-1}$ = max |a_i|, i = 0, …, n-1;**

$a_{1,n}$ = max |a_i|, i = 1, …, n.

То отримаємо: $a_{0,3} = 9$; $a_{1,4} = 9$.

Тоді: $\frac{|a_0|}{a_{1,4} + |a_0|} = \frac{3}{7+3} \le |x| \le \frac{|a_4| + a_{0,3}}{|a_4|} = \frac{7+3}{3}$;

$7/16 \le |x| \le 10$

Теорема 2:

Оскільки **a_ =** $\max_i |a_i|$, **$a_i < 0$; m = max i :$a_i < 0$.**

То отримаємо: **a_ =** 9**; m =** 3;

Тоді: **а)верхня межа додатних коренів** $x^+ \le 1 + \sqrt[4-3]{\frac{9}{1}} = 10$;

б)нижня межа додатних коренів: x:=1/y;

$7y^4 + 3y^3 + 9y^2 + 2x - 1 = 0$;

a_ = 1, **m** = 0; $y^+ = \frac{1}{x^+} \le 1 + \sqrt[4-0]{\frac{1}{7}} = 1.61478$;

$x^+ \ge 0.619279$;

в)нижня межа від'ємних коренів: x:=-x;

$x^4 + 2x^3 - 9x^2 + 3x - 7 = 0$;

a_ = 9, **m** = 2; $-x^{-} \leq 1 + \sqrt[4-2]{\frac{9}{1}}$;

$x^{-} \geq -4$

г)верхня межа від'ємних коренів: $x := -\frac{1}{y}$;

$$7y^4 - 3y^3 + 9y^2 - 2y - 1 = 0;$$

a_ = 3, **m** = 3; $y^{+} = -\frac{1}{x^{-}} \leq 1 + \sqrt[4-3]{\frac{3}{7}} = 1.42857$;

$x^{-} \leq -0{,}7$;

Теорема Штурма. Будуємо послідовність Штурма:

$f_0 = x^4 - 2x^3 - 9x^2 - 3x - 7$;

$f_1 = 4x^3 - 6x^2 - 18x - 3$;

$f_2 = \frac{1}{8}(-42x^2 + 36x + 59)$;

$f_3 = \frac{1}{147}(2284x + 1506)$;

$f_4 = const > 0$

	-4	-0.7	0	0,6	10
f_0	+	–	–	–	+
f_1	–	+	–	–	+
f_2	–	+	+	+	+
f_3	–	+	+	+	+
f_4	+	+	+	+	+
	2	1	1	1	0

Отже, корені належать проміжкам (-4; -0.7) та (0.6; 10)

Результати роботи програми

Результати роботи програми зображено на рис.1.

```
iter =  18
x1 = -2.1778
iter2 =  10
x2 =  4.3268
iter3 =  3
x3 = -2.1778
iter =  19
x4 =  4.3268
iter2 =  10
x5 =  4.3268
iter3 =  4
x6 =  4.3268
iter3 =  7
x7 = -2.1778
>>
```

Рис. 1.

Висновок

Під виконання лабораторної роботи було визначено, що рівняння зазначене умовою роботи має один дійсний корінь на інтервалі (0,4; 2). За допомогою наближених методів пошуку кореня рівняння, а саме: бісекції, методу хорд та методу Ньютона, було знайдено розв’язок рівняння з точністю 0,00001. Згідно результатів роботи програми, встановлено, що найшвидшим є метод Ньютона, потім слідує метод бісекції та на останньому місці – метод хорд.

Лістинг програми

Головна функція програми

```
x1 = findroot(-4, -0.7, 0.00001)

x2 = findroot2(-4, -0.7, 0.00001)

x3 = findroot3(-2, 0.00001)

x4 = findroot(0.619279, 10, 0.00001)
```

```
x5 = findroot2(0.619279, 10, 0.00001)

x6 = findroot3(5, 0.00001)

x7 = findroot3(-5, 0.00001)
```

Функція, що містить похідну поліному

```
function val = dpol(x)
  val = 4*x^3 - 6*x^2 - 18*x - 3;


endfunction
```

Функція, що містить поліном

```
function val = pol (x)
val = 0;
val = x^4 - 2*x^3 - 9*x^2 - 3*x - 7;
endfunction
```

Реалізація методів

```
function x = findroot (a, b, eps);
if pol(a) == 0
  x = a;
  return;
endif
if pol(b) == 0
  x = b;
  return;
endif
c = (b-a)/2;
iter = 0;
while (b-a)/2 > eps
  iter++;
  c = (b-a)/2;
  mid = a + c;
  if sign(pol(a)) != sign(pol(mid))
    b = mid;
```

```
   else a = mid;
  endif
endwhile
x = mid;
iter
endfunction
function x = findroot2 (a, b, eps);
if pol(a) == 0
  x = a;
  return;
endif
if pol(b) == 0
  x = b;
  return;
endif
iter2 = 0;
c = (a*pol(b) - b*pol(a))/(pol(b) - pol(a));
while abs(pol(b) - pol(a)) > eps
   a = b - (b - a) * pol(b) / (pol(b) - pol(a));
   b = a + (a - b) * pol(a) / (pol(a) - pol(b));
  iter2++;
endwhile
x = b;
iter2
endfunction
function x = findroot3(x0, eps);
  x1 = x0 - (pol(x0)/dpol(x0));
  iter3 = 0;
while abs(x1 - x0) > eps
  x0 = x1;
  x1 = x1 - (pol(x1)/dpol(x1));
```

```
  iter3++;
endwhile
x = x1;
iter3
endfunction
```