m=3

Інтервали	L=0	L=1	L=2	L=3
[-1,1]	i=10 t=64	i=9 t=59	i=8 t=53	i=8 t=55
[-10,10]	i=10 t=64	i=9 t=59	i=8 t=53	i=8 t=54
[-100,100]	i=11 t=71	i=9 t=59	i=8 t=52	i=8 t=55

З отриманих результатів видно, що збіжність схеми (1.1) суттєво залежить від вибору параметрів *l* та *m*. При збільшенні параметрів *m* та *l* ми хочемо прискорити збіжність методу, але існують такі оптимальні значення m_0 і l_0, при збільшенні яких збіжність не покращується. Для даного прикладу $m_0=2$, $l_0=2$.

1. Венгерський П.С., Трушевський В.М., Сеньо П.С. Використання інтерваль-ного аналізу для побудови одного типу двосторонніх методів для розв'язування систем нелінійних рівнянь.Львів,1998.-24с.-Рукопис деп. в УкрНДІНТІ(в друці).
2. Давиденко Д.Е. О приложении метода вариации параметра к теории нелинейных функциональных уравнений// Укр.мат.журн. 1955., Т.7. №1.

УДК 517.6

В.Д. Вовк, Р.Б. Петришин, Г.А. Шинкаренко

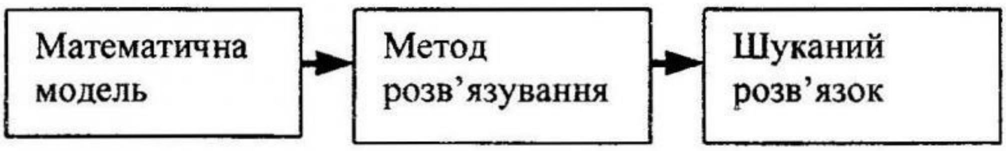
ВЕРХІВКА СИСТЕМИ КЛАСІВ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ ЧИСЕЛЬНИХ МЕТОДІВ

Застосування об'єктного підходу не має альтернативи при розробці складних програмних систем. Оскільки складність реалізації окремих чисельних методів, наприклад, методу скінченних елементів (МСЕ), сумнівів не викликає, то зрозумілими є намагання розробників використовувати засоби об'єктно-орієнтованого програмування при створенні своїх систем. Проте у більшості випадків це призводить до збільшення в загальному затрат на створення програмного продукту, а ефективність програми не тільки не зростає, а часто й зменшується. Причиною цього є те, що видима складність структури математичних методів належить до іншої категорії, ніж структурна складність великих інформаційних систем, при створенні яких об'єктний підхід дає максимальний виграш. Одним із підтверджень сказаного є відомий факт, що питома вага стандартних операцій лінійної алгебри в реалізаціях чисельних методів становить понад 70 відсотків.

Все ж використання об'єктного підходу при чисельному аналізі математичних моделей може бути ефективним, а у багатьох випадках – і необхідним. Йдеться про створення структурно-складних програмних комплексів, здатних проводити дослідження декількох математичних моделей багатьма методами. Тільки у цьому випадку накладні видатки на підтримку об'єктного підходу окуповуються багаторазовим використанням і гнучкістю раніше написаного програмного коду.

Проте розробка окремої системи класів для кожної конкретної, нехай навіть досить загальної, задачі апріорі обмежує область використання створеного програмного забезпечення. Подальше ж його адаптування до застосувань, що не входять у наперед визначене коло задач, тим більше не ефективне, чим більший радіус цього кола. У даній роботі розглядається ідея створення єдиної для всіх чисельних методів ієрархії класів, реалізація і підтримка якої іншими розробниками програмного забезпечення дала б можливість використати переваги об'єктного підходу повною мірою. Нижче запропоновано варіант верхівки повної структури класів чисельного аналізу з прикладом деталізації однієї гілки до рівня МСЕ.

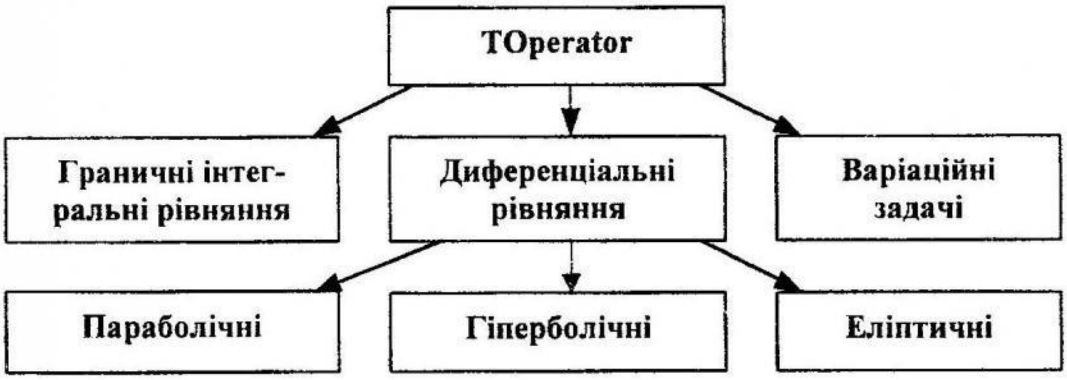
У вершині ієрархії лежать три базові класи



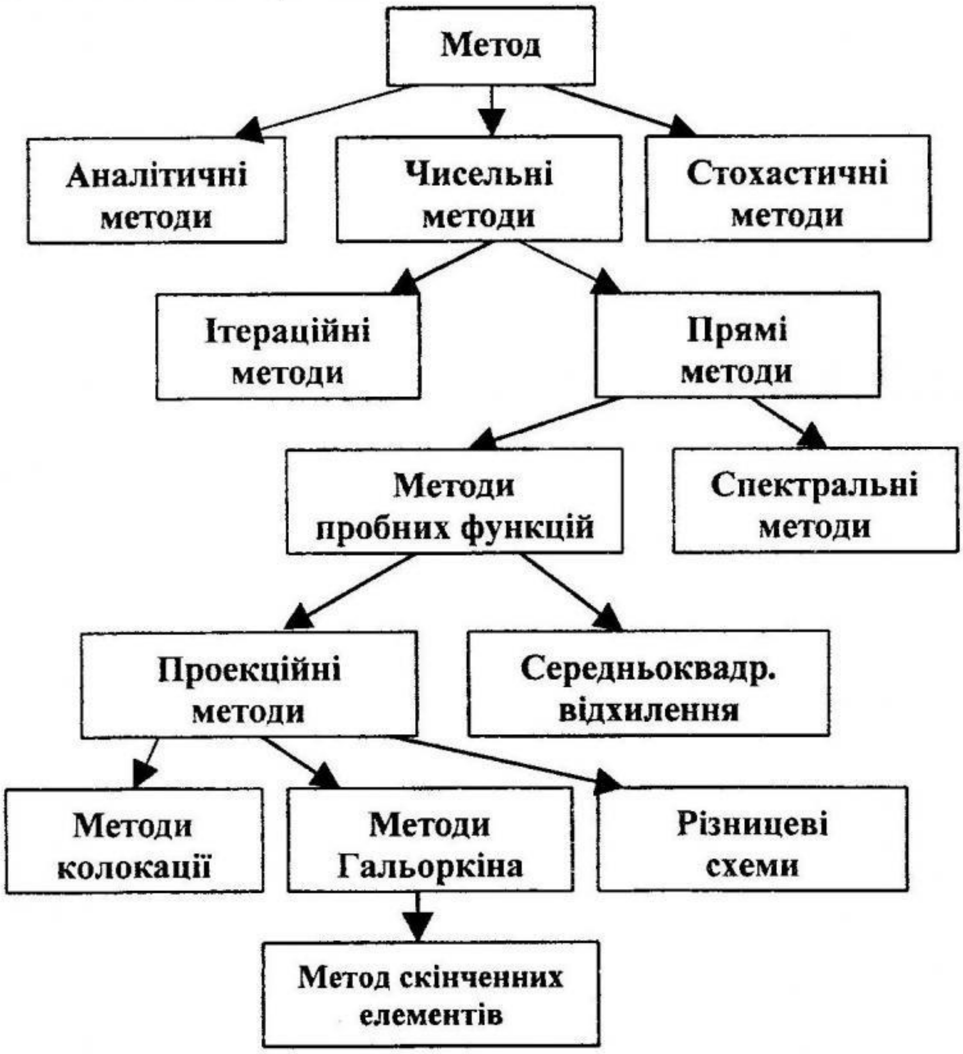
Наведемо конкретний вигляд на мові С++ тільки першого:

```
Class   TMathModel
        {TOperator   *F, *G;        // Рівняння моделі і крайових умов
        TFunctional  * f,  *g;      // Дані математичної моделі
        TGeometry   * Г; //Розглядувана область визначення розв'язку
        TmathModel ();              // Конструктор
        }
```

Зокрема, тип **TOperator** є вершиною розгалуженої системи класів:



Клас методів має три основні гілки:



Зрозуміло, що наведена тут ієрархія класів не є ні достатньо повною, ні достатньо детальною. Зокрема, класи **Ітераційні методи, Спектральні методи** лежать в основі розгалужених далі гілок методів. Проте наведений вибір кожного пункту даної ієрархії має достатнє математичне обґрунтування і може служити початковим наближенням для створення загальноприйнятої структури класів чисельних методів.

Зрозуміло, що для покриття затрат на створення такої системи класів (як і будь-якої іншої програмної системи) потрібна певна "критична маса" організаційних заходів, програмних напрацювань і користувачів. Проте приклади використання відомих структур класів (наприклад, MFC фірми Microsoft) свідчать про необхідність та ефективність такої розробки.

УДК 519.6

В.Д. Вовк, Р.Б. Петришин, Г.А. Шинкаренко

РОЗВ'ЯЗУВАНІСТЬ ВАРІАЦІЙНОЇ ЗАДАЧІ УЗАГАЛЬНЕНОЇ ТЕРМОПРУЖНОСТІ

1. Постановка початково-крайової задачі. Для дослідження закономірностей фізичних процесів у пружних тілах, де враховується скінченна швидкість поширення тепла, розглянемо початково-крайову задачу узагальненої термопружності, сформульовану Підстригачем та Коляно.

2. Варіаційна задача. Ввівши простори допустимих функцій та простори лінійних неперервних функціоналів, можемо сформулювати відповідну варіаційну задачу узагальненої термопружності:

$$
\begin{cases}
\text{задано}\, u_0 \in V, v_0 \in H, \theta_0 \in Z, q_0 \in R \square \in (l,\mu) \in L^2(0,T;V'\times G'); \\
\text{знайти трійку}\, \psi = (u,\theta,q) \in L^2(0,T;V\times G\times R)\ \text{таку, що} \\
m(u''(t),v) + a(u'(t),v) + c(u(t),v) - b(\theta(t),v) = \rangle l(t),v\langle, \\
s(\theta'(t),\xi) + d(q(t),\xi) + b(\xi,u'(t)) = \rangle \mu(t),\xi\langle, \\
\tau_r k(q'(t),r) - d(r,\theta(t)) + k(q(t),r) = 0, \\
m(u'(0)-v_0,v) = 0, c(u(0)-u_0,v) = 0 \forall v \in V, \\
s(\theta(0)-\theta_0,\xi) = 0 k(q(0)-q_0,r) = 0 \forall \xi \in G, \forall r \in R.
\end{cases}
\qquad (2.1)
$$