Реферетивна частина

Опис роботи бібілотек <complex.h> та <tgmath.h> на Ci

Опис роботи бібліотеки <complex> на С++

Порівняння результатів написаного коду

Виконана студентом 2-ої групи 2-го курсу

спеціальності «Комп’ютерна математика»

Дряпікою Арсенієм Валентиновичем

***<Complex.h>***

complex.h — заголовний файл стандартної бібліотеки мови програмування С, в якому оголошуються функції для комплексної арифметики. Ці функції використовують вбудований тип complex, котрий з'явився у стандарті C99.

Макроси, пов’язані з <complex.h>

Деякі з макросів <complex.h> показані нижче. Значення в лівій частині описують макроси в complex.h, а в правій частині описують розширення цих макросів за допомогою ключових слів (\_Imaginary, \_Complex), доданих у стандарті C99.

| ***Macro Name*** | ***Expands To*** |
| --- | --- |
| *complex* | *\_Complex* |
| *imaginary* | *\_Imaginary* |
| *\_Complex\_I* | *(const float \_Complex) i* |
| *\_Imaginary\_I* | *(const float \_Imaginary) i* |

Функції, пов’язані з <complex.h>

Заголовний файл <complex.h> також містить деякі вбудовані функції для роботи з комплексним числом. Вони діляться на: тригонометричні, гіперболічні, експонентні та логарифмічні, степеневі та абсолютні, куруючі.

***Тригонометричні функції***

Функції повертають косинус комплексного числа:

* float complex ccosf(float complex arg)
* double complex ccos(double complex arg)
* long double complex ccosl(long double complex arg)

Функції повертають cинус комплексного числа:

* float complex casinf(float complex arg)
* double complex casin(double complex arg)
* long double complex casinl(long double complex arg)

Функції повертають тангенс комплексного числа:

* float complex catanf(float complex arg)
* double complex catan(double complex arg)
* long double complex catanl(long double complex arg)

***Гіперболічні функції***

Функції повертають значення гіперболічного косинуса комплексного числа:

* double complex ccosh(double complex arg)
* float complex ccoshf(float complex arg)
* long double complex ccoshl(long double complex arg)

Функції повертають значення гіперболічного синуса комплексного числа:

* double complex csinh(double complex arg)
* float complex csinhf(float complex arg)
* long double complex csinhl(long double complex arg)

Функції повертають значення гіперболічного тангенса комплексного числа:

* double complex ctanh(double complex arg)
* float complex ctanhf(float complex arg)
* long double complex ctanhl(long double complex arg)

***Експонентні та Логарифмічні функції***

Функції вираховують комплексну експоненту по застасуванню е:

* float complex cexpf(float complex arg)
* double complex cexp(double complex arg)
* long double complex cexpl(long double complex arg)

Функції повертають натуральний (основа е) логарифм від arg:

* double complex clog(double complex arg)
* float complex clogf(float complex arg)
* long double complex clogl(long double complex arg)

***Степеневі та Абсолютні функції***

Функції повертають абсолютне значення складного аргументу:

* float cabsf(float complex arg)
* double cabs(double complex arg)
* long double cabsl(long double complex arg)

Функції повертають значення а^b, де а і b комплексні числа:

* double complex cpow(double complex x, double complex y)
* float complex cpowf(float complex x, float complex y)
* long double complex cpowl(long double complex x, long double complex y)

Функції повертають комплексний квадратний корінь аргументу:

* float complex csqrtf(float complex arg)
* double complex csqrt(double complex arg)
* long double complex csqrtl(long double complex arg

***Керуючі функції***

Функції повертають фазовий кут комплексного аргументу(у радіанах):

* float cargf(float complex arg)
* double carg(double complex arg)
* long double cargl(long double complex arg)

Функції повертають уявну частину комплексного числа:

* float cimagf(float complex arg)
* double cimag(double complex arg)
* long double cimagl(long double complex arg)

Функції повертають спряжене комплексне число:

* float complex conjf(float complex arg)
* double complex conj(double complex arg)
* long double complex conjl(long double complex arg)

Функції повертають дійсну частину комплексного числа:

* float crealf(float complex arg)
* double creal(double complex arg)
* long double creall(long double complex arg)

***Перезавантажені оператори***

Всі оператори перевантажені для роботи з комплексними числами:

* operator =
* operator -
* operator +
* operator \*
* operator /
* operator \*=
* operator +=
* operator -=
* operator /=
* operator ==
* operator !=

**<tgmath.h>**

thmath.h – заголовний файл для виконання простих математичних операцій. Всі ці функції приймають double, якщо не визначено інакше. Для роботи з типами float та long double використовуються функції з постфіксами f та l відповідно.

***Базові функції***

* abs – повертає абсолютну величину цілого числа
* acos – арк-косинус
* asin – арк-синус
* atan – арк-тангенс
* atan2 – арк-тангес с двома параметрами
* ceil – окрулгення до найближчого цілого числа
* cos - косинус
* cosh – гіперболічний косинус
* exp – знаходження експоненти
* fabs – повертає абсолютну величну з плаваючою точкою
* floor – округление до найближчого меншого цілого
* fmod – заходження остачі від ділення
* frexp – розбиває число з точкою на мантису и показник степеня
* ldexp – множення числа з точкою на цілий ступінь двох
* log – натуральний логарифм
* log10 – логарифм з основою 10
* modf(x,p) – знаходить цілу та дробову частину числа з точкою
* pow(x, y) – результат піднесення до степені
* sin – синус
* sinh – гіперболічний синус
* sqrt – квадратний корінь
* tan – тангенс
* tanh – гіперболічний тангенс

**<complex>**

<complex> - стандартна бібліотека реалізує комплексний клас, що містить комплексні числа в декартовій форму і кілька функції перевантажень для роботи з ними.

***Комплексні значення***

* real – повертає дійсну частину комплексного числа
* imag – повертає уявну частину комплексного числа
* abs - повертає абсолютну величину цілого числа
* arg – повертає фазовий кут комплексного числа
* norm – повертає нормальне значення комплексного числа
* conj – повертає спряжене комплексне число
* polar – повертає комплексне число з полярних компонентів
* proj – повертає проекцію комплексного числа x на сферу Рімана

***Трансцендентальні перевантаження***

* cosh – гіперболічний косинус комплексного числа
* cos – гіперболічний косинус комплексного числа
* exp – знаходження експоненти комплексного числа
* log – натуральний логарифм комплексного числа
* log10 – логарифм з основою 10 комплексного числа
* pow – результат піднесення до степені комплексного числа
* sin – синус комплексного числа
* sinh – гіперболічний синус комплексного числа
* sqrt – квадратний корінь комплексного числа
* tan – тангенс комплексного числа
* tanh – гіперболічний тангенс комплексного числа
* acos – арк-косинус комплексного числа
* acosh – гіперболічний арк-косинус комплексного числа
* asin - арк-синус комплексного числа
* asinh – гіперболічний арк-синус комплексного числа
* atan - арк-тангенс комплексного числа
* atanh – гіперболічний арк-тангенс комплексного числа

***Перезавантажені оператори***

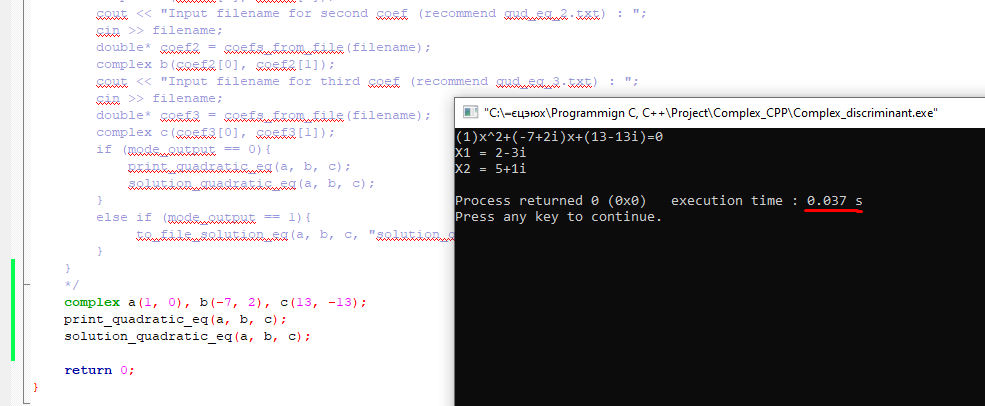
Всі оператори, перевантажені для роботи з комплексними числами:

* operator =
* operator -
* operator +
* operator \*
* operator /
* operator \*=
* operator +=
* operator -=
* operator /=
* operator ==
* operator !=

***Порівняння швидкодії знаходження коренів квадратного рівняння***

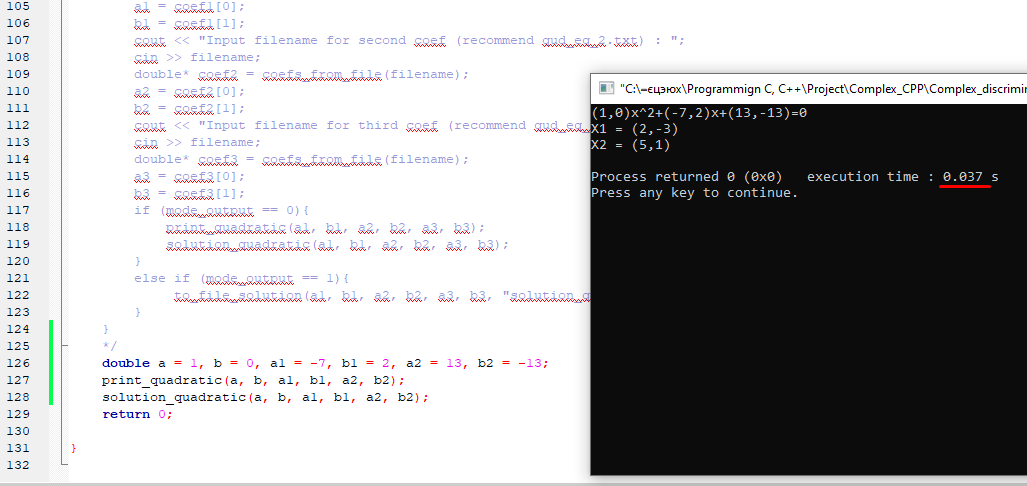
*Для чистоти експерименту коефіцієнти квадратного рівняння будуть однаковими і відразу задаються в основній частині коду.*

***Швидкодія знаходження коренів класом <complex> на С++, описаним мною***



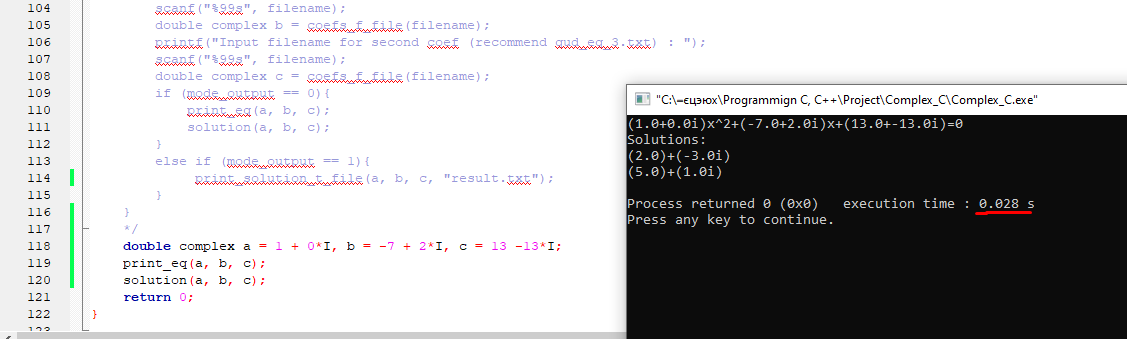
*Використаний час: 0.037 секунд.*

***Швидкодія знаходження коренів вбудованим класом <complex> на С++***



*Використаний час: 0.037 секунд.*

***Швидкодія знаходження коренів вбудованим класом <complex.h> та <tgmath.h> на Сi***



*Використаний час: 0.028 секунд.*

**Висновок**

Швидкодія роботи мого алгоритму дорівнює 0.037 секунд, як і швидкодія із вбудованою бібліотекою на С++. З цього можна зробити висновок, що описаний мною код оптимальний і працює не гірше, ніж вбудована бібліотека <complex> на С++. Швидкодія знаходження коренів квадратного рівняння на Сі з вбудованими бібліотеками становить 0.028 секунд, що є швидше ніж вбудована бібліотека на С++ і описаний мною алгоритм.