

КПІ ім. Ігоря Сікорського
Факультет інформатики та обчислювальної техніки
Кафедра інформатики та програмної інженерії

Звіт до комп'ютерного практикуму з курсу
“Основи програмування”

Прийняв доцент кафедри ІІІ
Лісовиченко О.І.

Виконав студент групи ІІ-21
Кулькова А. С.

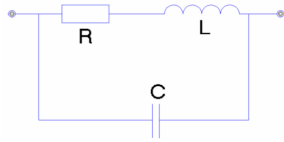
“2” січня 2022 р.

Київ 2022

Комп'ютерний практикум №8

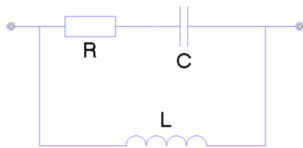
Тема: Застосування структур.

Завдання: Написати програму для обчислення комплексного опору заданого коливального контуру в залежності від частоти струму



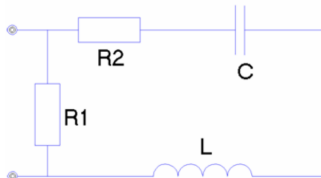
$$Z = \frac{\frac{L}{C} - i \frac{R}{\omega C}}{R + i(\omega L - \frac{1}{\omega C})}$$

б)



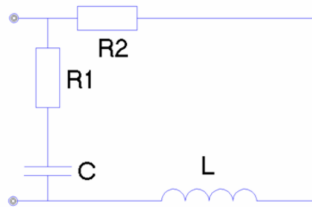
$$Z = \frac{\frac{L}{C} + i \frac{R}{\omega C}}{R + i(\omega L - \frac{1}{\omega C})}$$

в)



$$Z = \frac{R_1 R_2 + i R_1 (\omega L - \frac{1}{\omega C})}{R_1 + R_2 + i(\omega L - \frac{1}{\omega C})}$$

г)



$$Z = \frac{R_1 R_2 + \frac{L}{C} + i(\omega L R_1 - \frac{R_2}{\omega C})}{R_1 + R_2 + i(\omega L - \frac{1}{\omega C})}$$

Текст програми

```
#include <stdio.h>

#include <math.h>

#include <stdlib.h>

typedef struct

{

    float re;

    float im;

} complex;
```

```

float validation(float min, float max);

int int_validation(int minimum, int maximum);

void print_complex(complex number);

void circuit1 (float l, float c, float w, float r1, float r2);
void circuit2 (float l, float c, float w, float r1, float r2);
void circuit3 (float l, float c, float w, float R_first, float
R_second);

void circuit4 (float l, float c, float w, float R_first, float
R_second);

complex compl_division(complex arg1, complex arg2);

int main(){

    void (*circuit[4])(float, float, float, float, float) =
{circuit1, circuit2, circuit3, circuit4};

    const float min_v = 0;

    const float max_v = 10000;

    do
    {

        int choice, incorrect;

        float f_min, f_max, f0, f, df, L, C, W, R, R1, R2 = 1;

        //prompting the user

        printf("Choose the circuit [1; 4]  ");

        choice = int_validation (1, 4);

        printf("Enter L  ");

        L = validation(min_v, max_v);

        printf("Enter C  ");

        C = validation(min_v, max_v);

```

```

printf("enter R:  ");

R1 = validation(min_v, max_v);

if(choice == 3 || choice == 4)
{
    printf("enter R2:  ");

    R2 = validation(min_v, max_v);

}

// asking for the interval
do{

    incorrect = 0;

    printf("Enter f min:  ");

    f_min = validation(min_v, max_v);

    printf("Enter f max:  ");

    f_max = validation(min_v, max_v);

    if(f_min >= f_max)
    {

        incorrect =1;

        printf("f max (%f) should be greater than f min(%f)\n
Please try again\n", f_max, f_min);

    }

} while(incorrect);

printf("Enter df:  ");

df = validation(min_v, max_v);

f0 = ( 1 / ( 2 * M_PI * sqrtf(L * C))); // значення
резонансної частоти

```

```

printf("\nresonant frequency = %f\n\n", f0);

f = f_min;

int i = 1;

do{

    printf("f%d = %lf ", i, f);

    W = 2 * M_PI * f;

    printf("Z%d = ", i);

    (*circuit[choice-1])(L, C, W, R1, R2);

    f += df;

    i++;

} while ( f < f_max);

printf("f%d = %lf ", i, f_max);

printf("Z%d = ", i);

W = 2 * M_PI * f_max;

(*circuit[choice-1])(L, C, W, R1, R2);

fflush(stdin);

printf("\nTry again? -- press enter (otherwise press any
other key)");

} while(getchar()=='\n');

return 0;
}

float validation(float min, float max)
{

    char error, end; float d;

    do{

        error = 0;

        if(scanf("%f%c", &d, &end)!=2 || end!='\n' || d <= min || d

```

```

> max)

    {

        fflush(stdin);

        printf("Not correct. Please enter a float number in
range (%f; %f]\n Enter the number again: ", min, max);

        error =1;

    }

} while(error);

return (d);
}

int int_validation(int minimum, int maximum)
{

    char error, end; int k;

    do{

        error = 0;

        if(scanf("%d%c", &k, &end)!=2 || end!='\n' || k < minimum
|| k > maximum)

        {

            fflush(stdin);

            printf("Not correct. Please enter an integer in range
[%d; %d]\n Enter the number again: ", minimum, maximum);

            error =1;

        }

    } while(error);

    return (k);
}

void print_complex(complex number)

```

```

{

    if(number.im < 0)

    {
        if(number.re < 1e-6) printf("%e ",number.re);

        else printf("%f ",number.re);

        if(fabs(number.im) < 1e-6) printf("- i %e\n",
fabs(number.im));

        else printf("- i %f\n", fabs(number.im));

    }

    else

    {

        if(number.re < 1e-6) printf("%e ",number.re);

        else printf("%f ",number.re);

        if(number.im < 1e-6) printf("+ i %e\n", number.im);

        else printf("+ i %f\n", number.im);

    }

}

void circuit1 (float l, float c, float w, float r1, float r2)
{

    complex z1, z2, answ;

    z1.re = 1/c;

    z1.im = - r1 / (w*c); //if z1.im < 0 --> in printf complex
print re - i im

    z2.re = r1;

    z2.im = (w * l - (1 / (w * c)));

```

```

    answ = compl_division(z1, z2);

    print_complex(answ);
}

void circuit2 (float l, float c, float w, float r1, float r2)
{
    complex q1, q2, answ;

    q1.re = 1/c;

    q1.im =  r1 / (w*c);

    q2.re = r1;

    q2.im = (w * l - (1 / (w * c)));

    answ = compl_division(q1, q2);

    print_complex(answ);
}

void circuit3 (float l, float c, float w, float R_first, float
R_second)
{
    complex s1, s2, answ;

    s1.re = R_first * R_second;

    s1.im =  R_first * (w * l - (1 / (w * c)));

    s2.re = R_first + R_second;

    s2.im = (w * l - (1 / (w * c)));

    answ = compl_division(s1, s2);

    print_complex(answ);
}

void circuit4 (float l, float c, float w, float R_first, float
R_second)

```



```

{
    complex v1, v2, answ;

    v1.re = R_first * R_second + 1 / c;
    v1.im = w * l * R_first - (R_second / (w * c));

    v2.re = R_first + R_second;
    v2.im = (w * l - (1 / (w * c)));

    answ = compl_division(v1, v2);

    print_complex(answ);
}

complex compl_division(complex arg1, complex arg2)
{
    complex div_result;

    div_result.re = (arg1.re * arg2.re + arg1.im * arg2.im) /
    (arg2.re * arg2.re + arg2.im * arg2.im);

    div_result.im = (arg1.im * arg2.re - arg1.re * arg2.im) /
    (arg2.re * arg2.re + arg2.im * arg2.im);

    return div_result;
}

```

Введені та одержані результати, теоретичні розрахунки

Валідація вводу даних

```

Try again. Please press enter (otherwise press any other key)
Choose the circuit [1; 4] -
Not correct. Please enter an integer in range [1; 4]
Enter the number again: 0
Not correct. Please enter an integer in range [1; 4]
Enter the number again: 2
Enter L 0
Not correct. Please enter a float number in range (0.000000; 10000.000000]
Enter the number again: 999999999999999
Not correct. Please enter a float number in range (0.000000; 10000.000000]
Enter the number again: 2
Enter C -1
Not correct. Please enter a float number in range (0.000000; 10000.000000]
Enter the number again: 1
enter R: -9
Not correct. Please enter a float number in range (0.000000; 10000.000000]
Enter the number again: 4

```

валідація частоти

```

Enter f min: 9
Enter f max: 2
f max (2.000000) should be greater than f min(9.000000)
Please try again
Enter f min: 2
Enter f max: 9
Enter df: 0
Not correct. Please enter a float number in range (0.000000; 10000.000000]
Enter the number again: 2

```

Контур 1

```

Choose the circuit [1; 4] 1
Enter L 1
Enter C 1
enter R: 4
Enter f min: 1
Enter f max: 3
Enter df: 2

resonance frequency = 0.159155

f1 = 1.000000   Z1 = 0.001894 - i 0.162054
f2 = 3.000000   Z2 = 0.000030 - i 0.053195

```

$$\frac{1}{2\pi\sqrt{1}} = \frac{1}{2\pi} \quad (\text{Decimal: } 0.15915\dots)$$

$$\frac{1 - i \frac{4}{2 \cdot 3.14}}{4 + i \left(2 \cdot 3.14 - \frac{1}{2 \cdot 3.14} \right)} = 0.00189\dots - 0.16213\dots i$$

$$\frac{1 - i \frac{4}{2 \cdot 3.14 \cdot 3}}{4 + i \left(2 \cdot 3.14 \cdot 3 - \frac{1}{2 \cdot 3.14 \cdot 3} \right)} = 0.00003... - 0.05322...i$$

Контур 2

```
Choose the circuit [1; 4] 2
Enter L 2.5
Enter C 10.3
enter R: 23.3
Enter f min: 10.4
Enter f max: 15.5
Enter df: 5

resonance frequency = 0.031364

f1 = 10.400000 Z1 = 0.000415 - i 0.001427
f2 = 15.400000 Z2 = 0.000192 - i 0.000985
f3 = 15.500000 Z3 = 0.000189 - i 0.000979
```

$$\frac{1}{2\pi\sqrt{2.5 \cdot 10.3}} = 0.03136...$$

Z

$$\frac{\frac{2.5}{10.3} + i \frac{23.3}{2\pi 10.4 \cdot 10.3}}{23.3 + i \left(2\pi 10.4 \cdot 2.5 - \frac{1}{2\pi 10.4 \cdot 10.3} \right)} = 0.00041... - 0.00142...i$$

$$\frac{\frac{2.5}{10.3} + i \frac{23.3}{2\pi 15.4 \cdot 10.3}}{23.3 + i \left(2\pi 15.4 \cdot 2.5 - \frac{1}{2\pi 15.4 \cdot 10.3} \right)} = 0.00019... - 0.00098...i$$

$$\frac{\frac{2.5}{10.3} + i \frac{23.3}{2\pi 15.5 \cdot 10.3}}{23.3 + i \left(2\pi 15.5 \cdot 2.5 - \frac{1}{2\pi 15.5 \cdot 10.3} \right)} = 0.00018... - 0.00097...i$$

Контур 3

```

Choose the circuit [1; 4] 3
Enter L 0.9
Enter C 40
enter R: 13
enter R2: 37
Enter f min: 0.01
Enter f max: 0.9
Enter df: 0.8

resonance frequency = 0.026526

f1 = 0.010000    Z1 = 9.620158 - i 0.023073
f2 = 0.810000    Z2 = 9.648069 + i 0.306737
f3 = 0.900000    Z3 = 9.654601 + i 0.340224

```

f0 (резонансна частота)

$$\frac{1}{2\pi\sqrt{0.9 \cdot 40}} = 0.02652\dots$$

Z

$$\frac{13 \cdot 37 + i13\left(2\pi 0.01 \cdot 0.9 - \frac{1}{2\pi 0.01 \cdot 40}\right)}{13 + 37 + i\left(2\pi 0.01 \cdot 0.9 - \frac{1}{2\pi 0.01 \cdot 40}\right)} = 9.62015\dots - 0.02307\dots i$$

$$\frac{13 \cdot 37 + i13\left(2\pi 0.81 \cdot 0.9 - \frac{1}{2\pi 0.81 \cdot 40}\right)}{13 + 37 + i\left(2\pi 0.81 \cdot 0.9 - \frac{1}{2\pi 0.81 \cdot 40}\right)} = 9.64806\dots + 0.30673\dots i$$

$$\frac{13 \cdot 37 + i13\left(2\pi 0.9 \cdot 0.9 - \frac{1}{2\pi 0.9 \cdot 40}\right)}{13 + 37 + i\left(2\pi 0.9 \cdot 0.9 - \frac{1}{2\pi 0.9 \cdot 40}\right)} = 9.65460\dots + 0.34022\dots i$$

```

Choose the circuit [1; 4] 4
Enter L 0.9
Enter C 40
enter R: 13
enter R2: 37
Enter f min: 0.01
Enter f max: 0.9
Enter df: 0.8

resonance frequency = 0.026526

f1 = 0.010000    Z1 = 9.621911 - i 0.214047
f2 = 0.810000    Z2 = 9.648301 + i 0.304358
f3 = 0.900000    Z3 = 9.654833 + i 0.338079

```

f0 (резонансна частота)

$$\frac{1}{2\pi\sqrt{0.9 \cdot 40}} = 0.02652\dots$$

Z

$$\frac{13 \cdot 37 + \frac{0.9}{40} + i \left(2\pi 0.01 \cdot 0.9 \cdot 13 - \frac{37}{2\pi 0.01 \cdot 40} \right)}{13 + 37 + i \left(2\pi 0.01 \cdot 0.9 - \frac{1}{2\pi 0.01 \cdot 40} \right)} = 9.62191\dots - 0.21404\dots i$$

$$\frac{13 \cdot 37 + \frac{0.9}{40} + i \left(2\pi 0.81 \cdot 0.9 \cdot 13 - \frac{37}{2\pi 0.81 \cdot 40} \right)}{13 + 37 + i \left(2\pi 0.81 \cdot 0.9 - \frac{1}{2\pi 0.81 \cdot 40} \right)} = 9.64830\dots + 0.30435\dots i$$

$$\frac{13 \cdot 37 + \frac{0.9}{40} + i \left(2\pi 0.9 \cdot 0.9 \cdot 13 - \frac{37}{2\pi 0.9 \cdot 40} \right)}{13 + 37 + i \left(2\pi 0.9 \cdot 0.9 - \frac{1}{2\pi 0.9 \cdot 40} \right)} = 9.65483\dots + 0.33807\dots i$$

Вивід через експоненціальну форму при малих значеннях опору

```

37 physics
Choose the circuit [1; 4] 2
Enter L 3000
Enter C 800
enter R: 8000
Enter f min: 100
Enter f max: 220
Enter df: 10

resonant frequency = 0.000103

f1 = 100.000000 Z1 = 1.688656e-08 - i 0.000002
f2 = 110.000000 Z2 = 1.395588e-08 - i 0.000002
f3 = 120.000000 Z3 = 1.172684e-08 - i 0.000002
f4 = 130.000000 Z4 = 9.992121e-09 - i 0.000002
f5 = 140.000000 Z5 = 8.615668e-09 - i 0.000001
f6 = 150.000000 Z6 = 7.505213e-09 - i 0.000001
f7 = 160.000000 Z7 = 6.596385e-09 - i 0.000001
f8 = 170.000000 Z8 = 5.843169e-09 - i 0.000001
f9 = 180.000000 Z9 = 5.211966e-09 - i 0.000001
f10 = 190.000000 Z10 = 4.677779e-09 - i 0.000001
f11 = 200.000000 Z11 = 4.221697e-09 - i 9.947094e-07
f12 = 210.000000 Z12 = 3.829205e-09 - i 9.473431e-07
f13 = 220.000000 Z13 = 3.489009e-09 - i 9.042827e-07

```

Висновки

Отже, програма обраховує значення комплексного опору на заданих значенням частоти для будь-якого з чотирьох конутрів на вибір користувача. Також реалізовано функцію ділення двох комплексних чисел.