

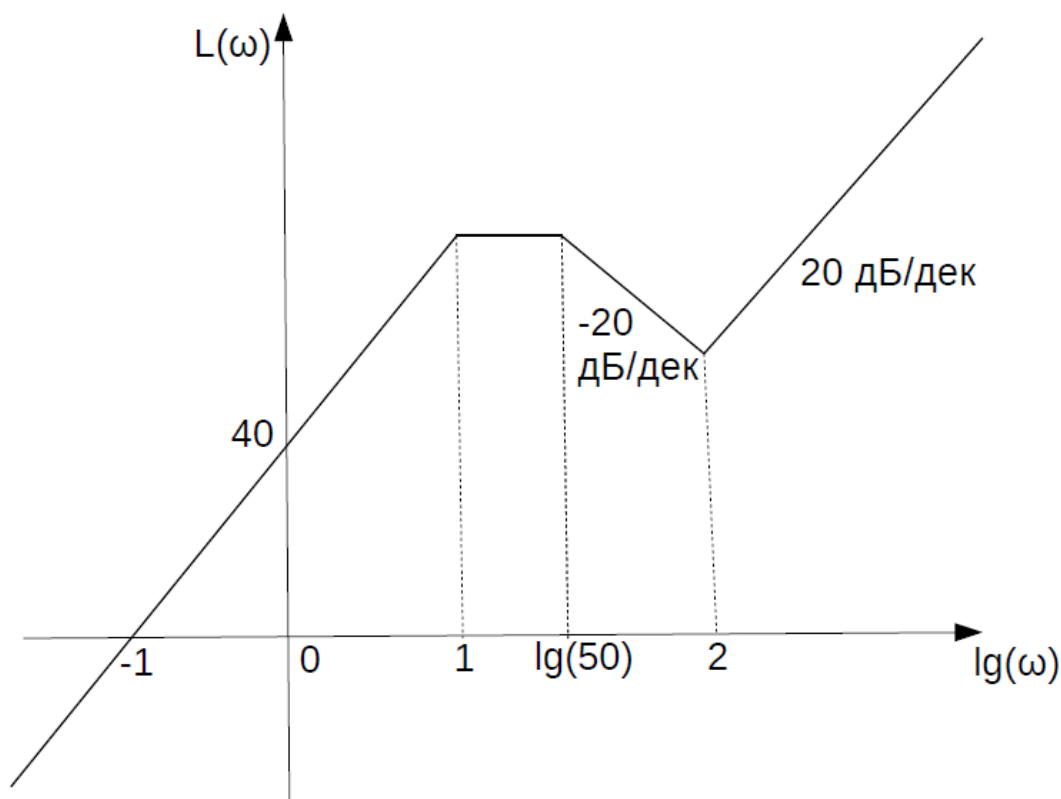
Время выполнения задания – 240 мин., язык – русский.

Максимальное количество баллов – 100.

Задача 1.

Задана асимптотическая логарифмическая амплитудно-частотная характеристика (ЛАЧХ) системы. Требуется:

1. Указать по виду асимптотической ЛАЧХ, на каких частотах система усиливает сигнал.
2. Найти сопрягающие частоты и соответствующие им постоянные времени.
3. Записать передаточную функцию для системы, соответствующую заданной ЛАЧХ, считая, что все элементарные звенья являются устойчивыми.
4. Указать, из каких типов элементарных звеньев состоит заданная система.
5. Составить дифференциальное уравнение системы с полученной передаточной функцией.
6. Исследовать устойчивость замкнутой и разомкнутой систем, если передаточная функция разомкнутой найдена в пункте 3.



Задача 2.

Построить график функции

$$y = \begin{vmatrix} x & x^2 & 1 \\ a & a^2 & 1 \\ b & b^2 & 1 \end{vmatrix}$$

при различных значениях параметров a и b .

Задача 3.

Рассмотрим задачу Коши для обыкновенного дифференциального уравнения первого порядка

$$(1) \quad du/dt = f(t, u), \quad u(t=0) = u_0$$

где $u(t)$ неизвестная функция действительного переменного t , а правая часть $f(t, u)$ есть известная функция двух переменных.

Для численного решения задачи Коши (1) вводится равномерная сетка $t_n = \tau n$, где $n=0, 1, 2, \dots$, и дифференциальное уравнение заменяется конечно-разностным уравнением на сеточную функцию u_n . Величину $\tau > 0$ назовем шагом сетки.

Рассмотрим уравнение (1) с правой частью вида

$$(2) \quad f(t, u) = \lambda u,$$

где $\lambda < 0$ – заданное действительное число.

Конечно-разностную схему назовем *абсолютно устойчивой* если решение конечно-разностного уравнения, u_n , соответствующего задаче Коши (1)-(2) является монотонно убывающей функцией n для любых значений $\tau > 0$. В противном случае – если при достаточно больших значениях τ решение теряет свойство монотонности либо растет неограниченно – схему назовем *условно устойчивой*.

Определить, является ли схема

$$(3) \quad (y_{n+1} - y_{n-1}) / 2\tau = (1/6) (f_{n+1} + 4f_n + f_{n-1})$$

абсолютно или условно устойчивой.

Задача 4.

На вход 2-х канальной системы обслуживания поступают заявки согласно пуассоновскому потоку с интенсивностью $a > 0$. Время обслуживания заявки на i -ом канале – случайная величина η_i , имеющая экспоненциальное распределение со средним значением $\frac{1}{b_i}$ ($b_i > 0, i = 1, 2$). Случайные величины η_1, η_2 независимы и не зависят от входного потока заявок. Предполагается, что «система с отказами», то есть, заявка, застающая оба канала занятыми, уходит из системы необслуженной, т.е. «теряется». Если оба канала свободны, то заявка выбирает канал для обслуживания с вероятностью $\frac{1}{2}$.

1. Построить марковский случайный процесс с непрерывным временем, описывающий работу системы. Обосновать марковское свойство.
2. Найти в установившемся режиме распределение числа заявок в системе.
3. Предположим, что $b_1 + b_2 = \beta$. Найти, при каких значениях b_1 и b_2 , удовлетворяющих этому условию, в установившемся режиме достигается минимум вероятности «потери заявки», то есть, вероятности того, что заявка уйдет из системы необслуженной.

Задача 5.

Определить значения переменных a , b , c , d , $a1$, $b1$, $c1$, а также содержимое массивов buf и $buf1$ после выполнения программы на языке программирования Си в UNIX-подобной операционной системе при условии, что файла $a.txt$ не существует в текущей директории. Что изменится при повторном запуске программы. Обосновать свое решение.

```
#include <stdio.h>
#include <unistd.h>
#include <sys/types.h>
#include <fcntl.h>
#include <string.h>
#include <stdlib.h>
#include <sys/wait.h>

int main()
{
    int a=4, b=4, c=4, d=4, a1=4, b1=4, c1=4, p[2];
    char buf[50], buf1[50];
    memset(buf,0,50); memset(buf1,0,50);
    close(1);
    pipe(p);
    creat("a.txt",600);
    close(0);
    dup(p[0]);
    if(fork()==0)
    {
        close(p[0]);
        dup(p[1]);
        printf("time\n");
        c=write(a,"data",5);
        a=open("a.txt",0);
        b=read(a,buf,2);
        d = write(1,"realy", 4);
        exit(2);
    }
    else
    {
        wait(&b1);
        close(p[1]);
        scanf("%s", buf);
        a1=read(p[0], buf1, 6);
        c1=open("a.txt",0);
    }
}
```