**МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**им. Н.Э. БАУМАНА**

Факультет: Информатика и системы управления Кафедра: Информационная безопасность (ИУ8)

**ТЕХНОЛОГИИ ОБЕСПЕЧЕНИЯ ИБ ОБЪЕКТОВ**

**Задание на тему:**

«Контроль защищенности речевой информации от утечки по акустическим (виброакустическим) техническим каналам»

Вариант 5

**Преподаватель:**

Медведев Н. В.

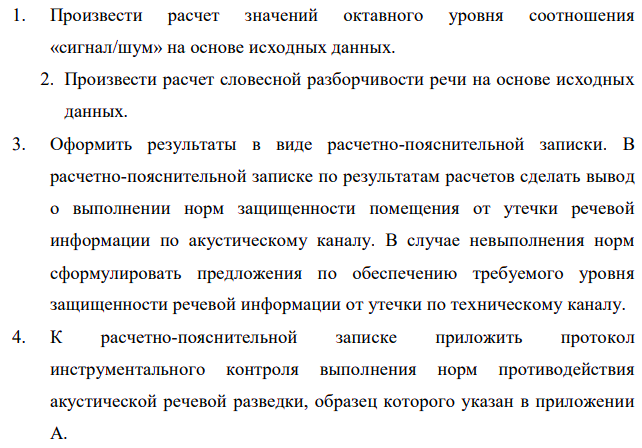
**Студент**:

Куликова А.В.

**Группа:**

ИУ8-31М

**Цель работы**



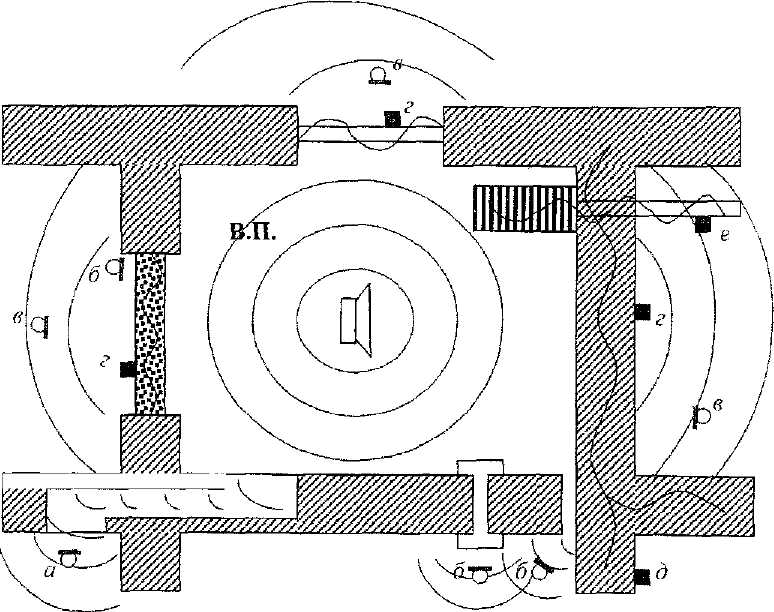
**Постановка задачи**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Вариант | Уровни речевого сигнала (дБ) | Уровни шумов (дБ) | Канал утечки информации  (на рис. 3) | Категория помещения | Язык речевого сигнала |
| **5** | 41, 38, 37, 34, 33 | 63, 58, 55, 52, 50 | а | ОН | русский |

**Примечание:** ОН – объект общего назначения.

*Акустические каналы*

а) Наибольшую опасность представляют технологические окна и каналы с большой площадью поперечного сечения, такие как короба коммуникаций и воздуховоды вентиляции. Эти объекты являются, по сути, акустическими волноводами и звуковые колебания могут распространяться по ним на значительные расстояния. Так, если поперечные размеры короба сравнимы с длиной звуковых волн, затухание, при распространении по нему звука, составляет δ = 0,01…1 дБ/м и зависит от размеров короба, материала стенок и пр.



**Ход работы**

Программная реализация на языке программирования Python.

from math import exp

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

# Функция для расчета затухания в коробах

def attenuation(length, size, material\_factor):

    # Расчет коэффициента затухания (delta) на основе размера

    delta = 0.01 + (0.99 \* size / 100)  # Примерная формула для затухания

    # Возвращает общее затухание, умножая delta на длину и коэффициент материала

    return delta \* length \* material\_factor

# Параметры частот для анализа

fs = [250, 500, 1000, 2000, 4000]  # Основные частоты

fs\_high = [355, 710, 1400, 2800, 5600]  # Высокочастотные параметры

fs\_low = [180, 355, 710, 1400, 2800]  # Низкочастотные параметры

# Функция для расчета W(R) в зависимости от R

def W(R):

    if R < 0.15:

        # Формула для W(R), если R меньше 0.15

        return 1.54 \* R \* (0.25) \* (1 - exp(-11 \* R))

    # Формула для W(R), если R больше или равен 0.15

    tmp = -11 \* R / (1 + 0.7 \* R)

    return 1 - exp(tmp)

# Функция для расчета суммы произведений ps и ks

def fR(ps, ks):

    return np.sum(ps \* ks)  # Возвращает сумму произведений элементов массивов ps и ks

# Функция для расчета ΔA(f) в зависимости от частоты f

def ΔA(f):

    if f > 1000:

        # Формула для ΔA(f), если f больше 1000 Гц

        return 1.37 + 1000 / (f \* 0.69)

    # Формула для ΔA(f), если f меньше или равен 1000 Гц

    return 200 / (f \* 0.43) - 0.37

# Функция для расчета p\_f(Qs)

def p\_f(Qs):

    Qs = list(Qs)  # Преобразуем Qs в список

    res = []  # Список для хранения результатов

    for qi in Qs:

        # Расчет временной переменной tmp на основе формулы p\_f

        tmp = (0.78 + 5.46 \* exp(-4.3 \* (10 \*\* (-3)) \* (27.3 - abs(qi) \*\* 2))) / \

              (1 + 10 \*\* (0.1 \* abs(qi)))

        if qi < 0:

            res.append(tmp)  # Если qi < 0, добавляем tmp в результаты

        else:

            res.append(1 - tmp)  # В противном случае добавляем 1 - tmp

    return res

# Функция для расчета k\_f(fs)

def k\_f(fs):

    return np.array(

        [

            2.57 \* (10 \*\* (-8)) \* fi \*\* (2.4)

            for fi in filter(lambda x: x <= 400, fs)  # Для частот <= 400 Гц

        ] + [

            1 - 1.074 \* exp(-10 \*\* (-4) \* fi \*\* 1.18)

            for fi in filter(lambda x: x > 400 and x < 10000, fs)  # Для частот между 400 и 10000 Гц

        ]

    )

def main():

    # Вывод формул для понимания расчетов

    print("\nФормулы:")

    print(r"""

        Формула для затухания:

            Δ = (0.01 + (0.99 \* size / 100)) \* length \* material\_factor

        Формула для W(R):

            W(R) =

            {

                1.54 \* R \* (0.25) \* (1 - e^(-11R)), если R < 0.15

                1 - e^(-11R / (1 + 0.7R)), иначе

            }

        Формула для fR(ps, ks):

            R = Σ(ps \* ks)

        Формула для ΔA(f):

            ΔA(f) =

            {

                1.37 + (1000 / (f \* 0.69)), если f > 1000

                (200 / (f \* 0.43)) - 0.37, иначе

            }

        Формула для p\_f(Qs):

            p\_f(Qs) =

            {

                (0.78 + (5.46 \* e^(-4.3 \* (10^(-3)) \* (27.3 - |qi|^2))) /

                (1 + 10^(0.1 \* |qi|)), если qi < 0

                иначе:

                1 - p\_f(Qs)

            }

        Формула для k\_f(fs):

            k\_f(fs) =

            {

                2.57 \* (10^(-8)) \* fi^(2.4), если fi <= 400

                1 - (1.074 \* e^(-10^(-4) \* fi^(1.18)), если fi > 400 и fi < 10000

            }

        """)

    # Уровни речи и шума в дБ

    speach\_lvls = np.array([41, 38, 37, 34, 33])

    noise\_lvls = np.array([63, 58, 55, 52, 50])

    dLt = 35

    # Расчет уровней Qs и Es на основе уровней речи и шума

    qs = speach\_lvls - noise\_lvls

    es = speach\_lvls - noise\_lvls - dLt

    # Создание таблицы для qi и Ei с использованием pandas DataFrame

    df\_qi\_ei = pd.DataFrame({

        'qi': qs,

        'Ei': es

    })

    print("Таблица qi и Ei:")

    print(df\_qi\_ei)

    # Расчет ΔA для каждой частоты в списке fs

    dAs = [ΔA(fsi) for fsi in fs]

    # Вывод формулы ΔA(f)

    for fsi, dAi in zip(fs, dAs):

        if fsi > 1000:

            formula = f"ΔA({fsi}) = 1.37 + (1000 / ({fsi} \* 0.69)) = {dAi:.2f}"

        else:

            formula = f"ΔA({fsi}) = (200 / ({fsi} \* 0.43)) - 0.37 = {dAi:.2f}"

        print(formula)

    # Расчет разности между k\_f для высоких и низких частот

    ks = k\_f(fs\_high) - k\_f(fs\_low)

    # Расчет Qs и ps на основе ранее вычисленных значений

    Qs = qs - dAs

    ps = p\_f(Qs)

    # Вывод формулы fR(ps, ks)

    r = fR(ps, ks)

    print(f"\nfR(ps, ks) = Σ(ps \* ks) = {r:.2f}")

    # Вывод формулы W(R)

    w = W(r)

    print(f"W(R) = {w:.4f}")

    # Пример расчета затухания для короба с заданными параметрами

    length = 10  # длина короба в метрах

    size = 50   # поперечное сечение в см^2

    material\_factor = 1.0   # коэффициент материала

    attn = attenuation(length, size, material\_factor)

    # Вывод формулы затухания с подставленными значениями

    delta = 0.01 + (0.99 \* size / 100)

    print(f"Затухание: Δ = ({delta:.2f}) \* {length} \* {material\_factor} = {attn:.2f} дБ")

    # Создание таблицы результатов с использованием pandas DataFrame

    df\_results = pd.DataFrame({

        'Частота (Гц)': fs,

        'ΔAi': np.round(dAs, 2),

        'Qi': np.round(Qs, 2),

        'Pi': np.round(ps, 3),

        'Ki': np.round(ks, 3)

    })

    print("\nТаблица результатов:")

    print(df\_results)

    # Визуализация результатов с использованием matplotlib

    plt.figure(figsize=(12, 6))

    plt.subplot(1, 2, 1)

    plt.plot(fs, dAs, marker='o', label='ΔAi')

    plt.plot(fs, Qs, marker='o', label='Qi')

    plt.title('ΔAi и Qi по частоте')

    plt.xlabel('Частота (Гц)')

    plt.ylabel('Значение')

    plt.legend()

    plt.subplot(1, 2, 2)

    plt.plot(fs, ps, marker='o', label='Pi')

    plt.plot(fs, ks, marker='o', label='Ki')

    plt.title('Pi и Ki по частоте')

    plt.xlabel('Частота (Гц)')

    plt.ylabel('Значение')

    plt.legend()

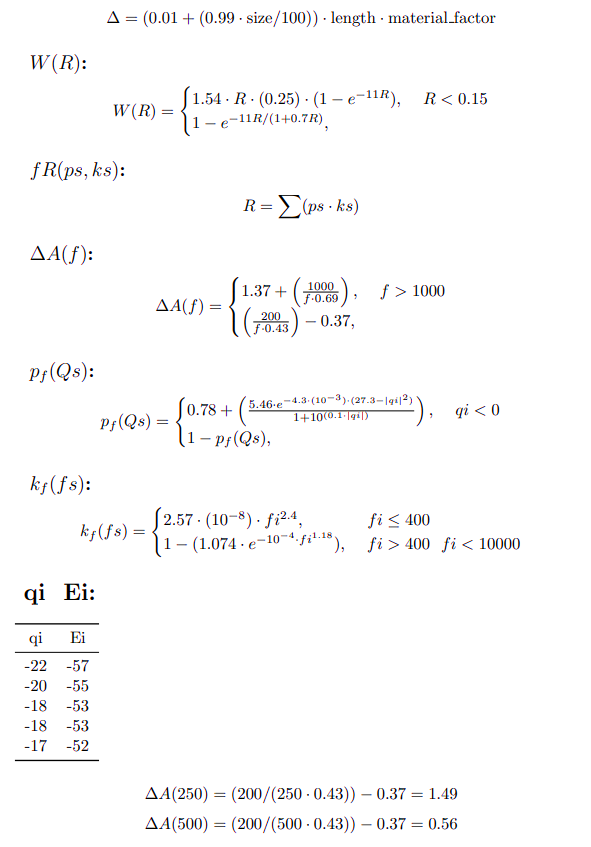
    plt.tight\_layout()

    plt.show()

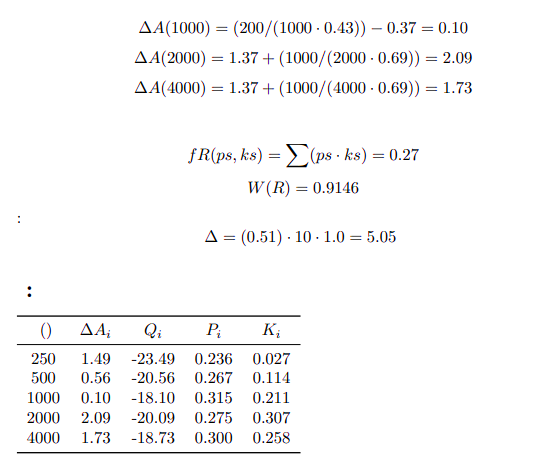
if \_\_name\_\_ == '\_\_main\_\_':

    main()

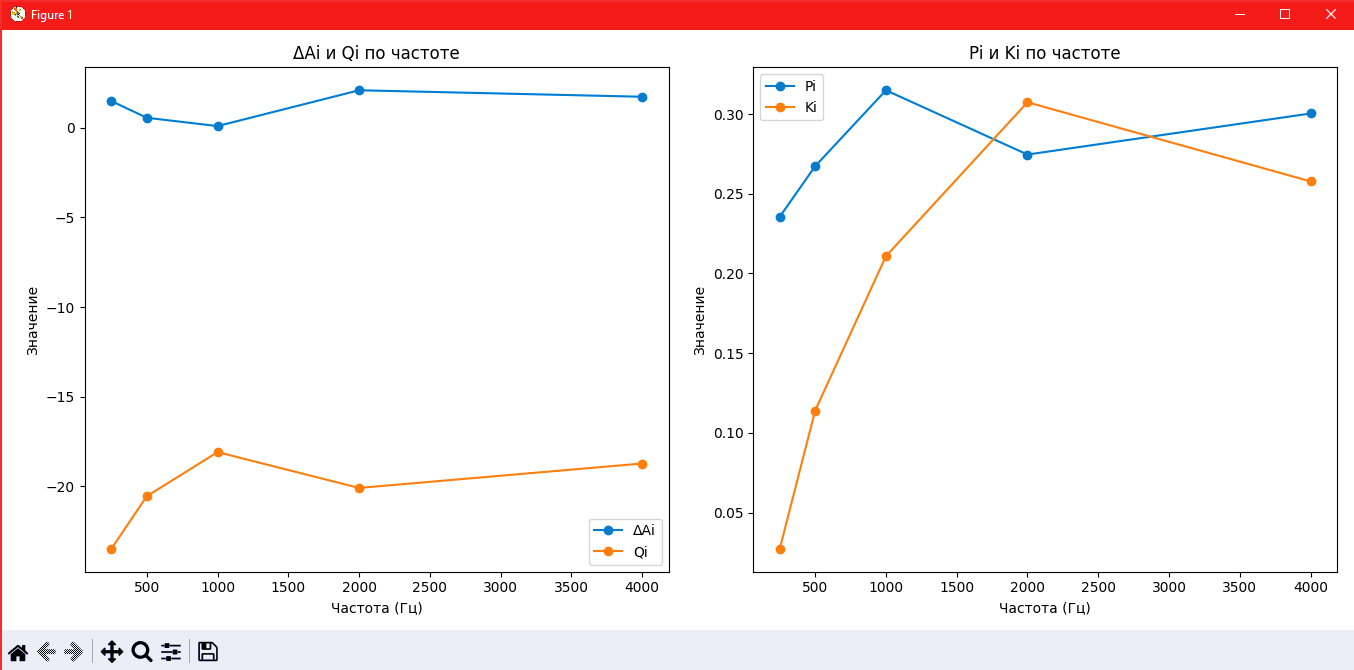
Результаты программной реализации представлены на рисунках 1 и 2, а также на рисунке 3 представлены графики по частотам.



1. Результаты программной реализации



1. Результаты программной реализации



1. Графики по частотам

Приложение а

Результат программной реализации:

C:\Users\USER\Desktop\test\pythonProject\.venv\Scripts\python.exe C:\Users\USER\Desktop\test\pythonProject\main.py

Формулы:

Формула для затухания:

Δ = (0.01 + (0.99 \* size / 100)) \* length \* material\_factor

Формула для W(R):

W(R) =

{

1.54 \* R \* (0.25) \* (1 - e^(-11R)), если R < 0.15

1 - e^(-11R / (1 + 0.7R)), иначе

}

Формула для fR(ps, ks):

R = Σ(ps \* ks)

Формула для ΔA(f):

ΔA(f) =

{

1.37 + (1000 / (f \* 0.69)), если f > 1000

(200 / (f \* 0.43)) - 0.37, иначе

}

Формула для p\_f(Qs):

p\_f(Qs) =

{

(0.78 + (5.46 \* e^(-4.3 \* (10^(-3)) \* (27.3 - |qi|^2))) /

(1 + 10^(0.1 \* |qi|)), если qi < 0

иначе:

1 - p\_f(Qs)

}

Формула для k\_f(fs):

k\_f(fs) =

{

2.57 \* (10^(-8)) \* fi^(2.4), если fi <= 400

1 - (1.074 \* e^(-10^(-4) \* fi^(1.18)), если fi > 400 и fi < 10000

}

Таблица qi и Ei:

qi Ei

0 -22 -57

1 -20 -55

2 -18 -53

3 -18 -53

4 -17 -52

ΔA(250) = (200 / (250 \* 0.43)) - 0.37 = 1.49

ΔA(500) = (200 / (500 \* 0.43)) - 0.37 = 0.56

ΔA(1000) = (200 / (1000 \* 0.43)) - 0.37 = 0.10

ΔA(2000) = 1.37 + (1000 / (2000 \* 0.69)) = 2.09

ΔA(4000) = 1.37 + (1000 / (4000 \* 0.69)) = 1.73

fR(ps, ks) = Σ(ps \* ks) = 0.27

W(R) = 0.9146

Затухание: Δ = (0.51) \* 10 \* 1.0 = 5.05 дБ

Таблица результатов:

Частота (Гц) ΔAi Qi Pi Ki

0 250 1.49 -23.49 0.236 0.027

1 500 0.56 -20.56 0.267 0.114

2 1000 0.10 -18.10 0.315 0.211

3 2000 2.09 -20.09 0.275 0.307

4 4000 1.73 -18.73 0.300 0.258

Process finished with exit code 0

приложение б

Отчет был реализован на LATEX.

\documentclass{article}

\usepackage{amsmath}

\usepackage{booktabs}

\begin{document}

\section\*{Формулы}

\subsection\*{Формула для затухания:}

\[

\Delta = (0.01 + (0.99 \cdot \text{size} / 100)) \cdot \text{length} \cdot \text{material\\_factor}

\]

\subsection\*{Формула для \( W(R) \):}

\[

W(R) =

\begin{cases}

1.54 \cdot R \cdot (0.25) \cdot (1 - e^{-11R}), & \text{если } R < 0.15 \\

1 - e^{-11R / (1 + 0.7R)}, & \text{иначе}

\end{cases}

\]

\subsection\*{Формула для \( fR(ps, ks) \):}

\[

R = \sum (ps \cdot ks)

\]

\subsection\*{Формула для \( \Delta A(f) \):}

\[

\Delta A(f) =

\begin{cases}

1.37 + \left( \frac{1000}{f \cdot 0.69} \right), & \text{если } f > 1000 \\

\left( \frac{200}{f \cdot 0.43} \right) - 0.37, & \text{иначе}

\end{cases}

\]

\subsection\*{Формула для \( p\_f(Qs) \):}

\[

p\_f(Qs) =

\begin{cases}

0.78 + \left( \frac{5.46 \cdot e^{-4.3 \cdot (10^{-3}) \cdot (27.3 - |qi|^2)}}{1 + 10^{(0.1 \cdot |qi|)}} \right), & \text{если } qi < 0 \\

1 - p\_f(Qs), & \text{иначе}

\end{cases}

\]

\subsection\*{Формула для \( k\_f(fs) \):}

\[

k\_f(fs) =

\begin{cases}

2.57 \cdot (10^{-8}) \cdot fi^{2.4}, & \text{если } fi \leq 400 \\

1 - (1.074 \cdot e^{-10^{-4} \cdot fi^{1.18}}), & \text{если } fi > 400 \text{ и } fi < 10000

\end{cases}

\]

\section\*{Таблица qi и Ei:}

\begin{tabular}{cc}

\toprule

qi & Ei \\

\midrule

-22 & -57 \\

-20 & -55 \\

-18 & -53 \\

-18 & -53 \\

-17 & -52 \\

\bottomrule

\end{tabular}

\vspace{1em}

\[

\Delta A(250) = (200 / (250 \cdot 0.43)) - 0.37 = 1.49

\]

\[

\Delta A(500) = (200 / (500 \cdot 0.43)) - 0.37 = 0.56

\]

\[

\Delta A(1000) = (200 / (1000 \cdot 0.43)) - 0.37 = 0.10

\]

\[

\Delta A(2000) = 1.37 + (1000 / (2000 \cdot 0.69)) = 2.09

\]

\[

\Delta A(4000) = 1.37 + (1000 / (4000 \cdot 0.69)) = 1.73

\]

\vspace{1em}

\[

fR(ps, ks) = \sum(ps \cdot ks) = 0.27

\]

\[

W(R) = 0.9146

\]

Затухание:

\[

\Delta = (0.51) \cdot 10 \cdot 1.0 = 5.05 \text{ дБ}

\]

\section\*{Таблица результатов:}

\begin{tabular}{ccccc}

\toprule

Частота (Гц) & $\Delta A\_i$ & $Q\_i$ & $P\_i$ & $K\_i$ \\

\midrule

250 & 1.49 & -23.49 & 0.236 & 0.027 \\

500 & 0.56 & -20.56 & 0.267 & 0.114 \\

1000 & 0.10 & -18.10 & 0.315 & 0.211 \\

2000 & 2.09 & -20.09 & 0.275 & 0.307 \\

4000 & 1.73 & -18.73 & 0.300 & 0.258 \\

\bottomrule

\end{tabular}

\end{document}