**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ**

**ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ П. О. СУХОГО**

Факультет автоматизированных и информационных систем

Кафедра «Информатика»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 1

по дисциплине **«Архитектура вычислительных систем»**

на тему: **«**Представление данных. Системы счисления**»**

Выполнил: студент гр. ИП-21

М. С. Есис

Принял: преподаватель

В. В. Брель

Дата сдачи отчета: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата допуска к защите: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата защиты: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Гомель 2024

**Цель работы**: ознакомиться с позиционными системами счисления, способами перевода данных из одной системы счисления в в другую, организацией арифметических операций над данными в позициионных системах.

**Задания к лабораторной работе:**

**Задание 1**. Необходимо написать программу, которая на входе получает две строки из ASCII-символов. В первой строке указывается основание системы счисления, во второй число. На выходе должно выводиться десятичное число, соответствующее введенному. В программе предусмотреть проверку соответствия цифр введенного числа алфавиту заданной системы счисления.

Основание системы счисления выбирается согласно порядковому номеру студента в журнале + 3. Вводимое число соответствует четырем первым цифрам зачетной книжки.

Текст программы:

def convert\_to\_decimal(base, number):

    for digit in number:

        if ord(digit) >= ord('A') + base:

            print("Ошибка: цифра {} не соответствует системе счисления!".format(digit))

            return None

    decimal = 0

    power = len(number) - 1

    for digit in number:

        if digit.isdigit():

            decimal += int(digit) \* (base \*\* power)

        else:

            decimal += (ord(digit) - ord('A') + 10) \* (base \*\* power)

        power -= 1

    return decimal

base = int(input("Введите основание системы счисления: "))

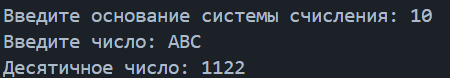
number = input("Введите число: ")

decimal = convert\_to\_decimal(base, number)

if decimal is not None:

    print("Десятичное число:", decimal)

Результат работы:



**Задание 2**. Необходимо написать программу, которая на входе получает две строки из ASCII-символов. В первой строке задается десятичное число, во второй указывается основание системы счисления, в которую необходимо перевести заданное число. На выходе должно выводиться число в новой системе счисления в виде строки из ASCII-символов.

Основание системы счисления выбирается согласно порядковому номеру студента в журнале + 3. Вводимое число соответствует четырем первым цифрам зачетной книжки.

Текст программы:

def convert\_number(decimal\_number, base):

    new\_number = ""

    while decimal\_number > 0:

        remainder = decimal\_number % base

        if remainder < 10:

            new\_number = str(remainder) + new\_number

        else:

            new\_number = chr(ord('A') + remainder - 10) + new\_number

        decimal\_number = decimal\_number // base

    return new\_number

decimal\_number = int(input("Введите десятичное число: "))

student\_number = int(input("Введите порядковый номер студента: "))

base = student\_number + 3

new\_number = convert\_number(decimal\_number, base)

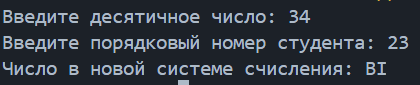
ascii\_number = ""

for digit in new\_number:

    ascii\_number += chr(ord('A') + int(digit))

print("Число в новой системе счисления:", ascii\_number)

Результат выполнения



**Задание 3.1.**Необходимо написать программу, которая на входе получает строку из ASCII-символов, состоящую из 0 и 1, соответствующей записи заданного двоичного числа. На выходе должно выводиться 2 строки ASCII-символов, соответствующие шестнадцатеричному и двоичному представлению введенного числа. В программе предусмотреть проверку ввода некорректных данных.

Вводимое число соответствует четырем первым цифрам зачетной книжки, переведенное в двоичное число с помощью калькулятора.

Текст программы:

def validate\_binary\_number(binary\_number):

    for digit in binary\_number:

        if digit != '0' and digit != '1':

            return False

    return True

def convert\_to\_hex(binary\_number):

    decimal\_number = int(binary\_number, 2)

    hex\_number = hex(decimal\_number).upper()[2:]

    return hex\_number

binary\_number = input("Введите двоичное число: ")

if not validate\_binary\_number(binary\_number):

    print("Ошибка: Введено некорректное двоичное число")

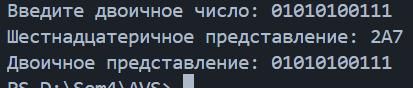
else:

    hex\_number = convert\_to\_hex(binary\_number)

    print("Шестнадцатеричное представление:", hex\_number)

    print("Двоичное представление:", binary\_number)

Результат выполнения:



**Задание 3.2.**Необходимо написать программу, которая на входе получает строку из ASCII-символов, соответствующей записи числа в заданной системе счисления (для нечетных вариантов – восьмеричная, для четных – шестнадцатеричная). На выходе должна выводиться строка ASCII-символов, состоящую из 0 и 1, соответствующая записи эквивалентного двоичного числа. В программе предусмотреть проверку ввода некорректных данных.

Вводимое число соответствует четырем первым цифрам зачетной книжки, переведенное в заданную систему счисления с помощью калькулятора.

Текст программы:

def convert\_to\_binary(number):

    try:

        decimal\_values = [ord(char) for char in number]

        binary\_number = ''.join([bin(value)[2:].zfill(8) for value in decimal\_values])

        return binary\_number

    except TypeError:

        return "Ошибка! Введены некорректные данные."

number = input("Введите число: ")

binary\_number = convert\_to\_binary(number)

print("Двоичная запись числа:", binary\_number)

Результат выполнения



**Задание 4.1**Написать программу, осуществляющую сложение и вычитание чисел в заданной системе счисления. В программе предусмотреть проверку ввода некорректных данных.

Основание системы счисления выбирается согласно порядковому номеру студента в журнале + 3.

Текст программы:

def add\_numbers(number1, number2, base):

    try:

        decimal\_number1 = int(number1, base)

        decimal\_number2 = int(number2, base)

        decimal\_sum = decimal\_number1 + decimal\_number2

        result = format(decimal\_sum, 'x' if base == 16 else 'o')

        return result

    except ValueError:

        return "Ошибка! Введены некорректные данные."

def subtract\_numbers(number1, number2, base):

    try:

        decimal\_number1 = int(number1, base)

        decimal\_number2 = int(number2, base)

        decimal\_difference = decimal\_number1 - decimal\_number2

        result = format(decimal\_difference, 'x' if base == 16 else 'o')

        return result

    except ValueError:

        return "Ошибка! Введены некорректные данные."

number1 = input("Введите первое число: ")

number2 = input("Введите второе число: ")

base = 11

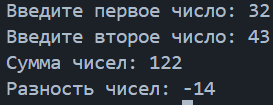
sum\_result = add\_numbers(number1, number2, base)

print("Сумма чисел:", sum\_result)

difference\_result = subtract\_numbers(number1, number2, base)

print("Разность чисел:", difference\_result)

Результат выполнения



**Задание 4.2.**Написать программу, осуществляющую умножение и деление чисел в заданной системе счисления. В программе предусмотреть проверку ввода некорректных данных.

Основание системы счисления выбирается согласно порядковому номеру студента в журнале + 3.

Текст программы:

def multiply\_numbers(number1, number2, base):

    try:

        decimal\_number1 = int(number1, base)

        decimal\_number2 = int(number2, base)

        decimal\_product = decimal\_number1 \* decimal\_number2

        result = format(decimal\_product, "x" if base == 16 else "o")

        return result

    except ValueError:

        return "Ошибка! Введены некорректные данные."

def divide\_numbers(number1, number2, base):

    try:

        decimal\_number1 = int(number1, base)

        decimal\_number2 = int(number2, base)

        decimal\_quotient = decimal\_number1 // decimal\_number2

        result = format(decimal\_quotient, "x" if base == 16 else "o")

        return result

    except ValueError:

        return "Ошибка! Введены некорректные данные."

    except ZeroDivisionError:

        return "Ошибка! Деление на ноль."

number1 = input("Введите первое число: ")

number2 = input("Введите второе число: ")

base = 14

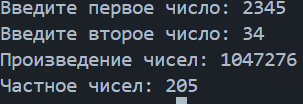
product\_result = multiply\_numbers(number1, number2, base)

print("Произведение чисел:", product\_result)

quotient\_result = divide\_numbers(number1, number2, base)

print("Частное чисел:", quotient\_result)

Результат выполнения



Вывод: ознакомилась с позиционными системами счисления, способами перевода данных из одной системы счисления в другую, организацией арифметических операций над данными в позиционных системах.