Міністерство освіти і науки України  
Національний технічний університет України  
«Київський політехнічний інститут»

Інститут Прикладного системного аналізу  
Кафедра Системного проектування

Лабораторна робота №2

з дисципліни «Архітектура обчислювальних систем»

«Дослідження роботи АЛУ»

Виконав:

студент групи ДА-81

Бондаренко Станіслав

Варіант №2

**Завдання:**

1. Власноруч представити константи, задані в таблиці 1, у форматі двійкових чисел з фіксованою крапкою в двійковому і шістнадцятирічному коді, а також в упакованому форматі десяткових чисел. Ознайомитись з синтаксисом введення чисельних констант в асемблері: EQU, 1.2, -12, 0123H, 0ABCDH, 01010b.
2. Пояснити призначення регістрів ЦПУ, що використовуються АЛУ.
3. Вивчити команди виконання арифметичних операцій над даними з фіксованою комою ADD, ADC, SUB, SBB, MUL, IMUL, DIV, IDIV, CWD, CWB, NEG, INC, DEC. Написати  програму, що ілюструє їх використання для заданих констант.
4. Представити A/B як десятковий дріб  з фіксованою комою. Навести приклад роботи з цим числом.
5. Вивчити команди виконання логічних операцій AND, OR, XOR, продемонструвати їх роботу за допомогою демонстраційної програми.
6. Вивчити команди виконання операцій над двійково-десятковими числами в упакованому форматі. Навести приклади використання команд DAA і DAS.
7. Порівняти результати роботи демонстраційних програм з результатами, отриманими власноруч.

**Варіант 2**

A = 223

B = -101

**Хід роботи**

**Завдання 1.** Власноруч представити константи, задані в таблиці 1, у форматі двійкових чисел з фіксованою крапкою в двійковому і шістнадцятирічному коді, а також в упакованому форматі десяткових чисел.

**Формат двійкових чисел з фіксованою крапкою** (якщо уявляємо, що фіксована крапка знаходиться у кінці разрядної сітки):

**223:** 0000 0000 1101 1111

**-101:**

101: 0000 0000 0110 0101

Інверсія: 1111 1111 1001 1010

+1: 1111 1111 1001 1011

**Формат двійкових чисел з фіксованою крапкою в шістнадцятирічному коді:**

**223:** 0000 0000 1101 1111 (двійк.)

0 0 13 15 (десятк.)

00DF (шіст.)

**-101:**  1111 1111 1001 1011 (двійк.)

15 15 9 11 (десятк.)

FF9B (шіст.)

**Упакований формат десяткових чисел:**

А = 22310 =0000 0010 0010 00112-10

B = -10110 =1000 0001 0000 00012-10

**Завдання 2.**

Процесор 8086 має 14 регістрів розрядністю по 16 біт. З них чотири регістри (AX, BX, CX, DX) - це регістри даних, кожен з яких окрім зберігання операндів і результатів операцій має ще і своє специфічне призначення:

* **регістр AX - множення, ділення, обмін з пристроями введення/виводу (команди введення і виводу);**
* **регістр BX - базовий регістр в обчисленнях адреси;**
* **регістр CX - лічильник циклів;**
* **регістр DX - визначення адреси введення/виводу.**

Для регістрів даних існує можливість роздільного використання обох байтів (наприклад, для регістра AX вони мають позначення AL - молодший байт і AH - старший байт).

Наступні чотири внутрішні регістри процесора - це сегментні регістри, кожен з яких визначає положення одного з робочих сегментів:

* **регістр CS (Code Segment) відповідає сегменту команд, що виконуються в даний момент;**
* **регістр DS (Data Segment) відповідає сегменту даних, з якими працює процесор;**
* **регістр ES (Extra Segment) відповідає додатковому сегменту даних;**
* **регістр SS (Stack Segment) відповідає сегменту стека.**

Наступні п'ять регістрів процесора

**SP** - Stack Pointer,

**BP** - Base Pointer,

**SI** - Source Index,

**DI** - Destination Index,

**IP** - Instruction Pointer

служать покажчиками (тобто визначають зміщення в межах сегменту). Наприклад, лічильник команд процесора утворюється парою регістрів CS і IP, а покажчик стека - парою регістрів SP і SS. Регістри SI, DI використовуються в строкових операціях, тобто при послідовній обробці декількох елементів пам'яті однією командою.

Останній регістр FLAGS - це регістр стану процесора (PSW). З його 16 розрядів використовуються тільки дев'ять:

**CF (Carry Flag)** - прапорець перенесення при арифметичних операціях,

**PF (Parity Flag)** - прапорець парності результату,

**AF (Auxiliary Flag)** - прапорець додаткового перенесення,

**ZF (Zero Flag)** - прапорець нульового результату,

**SF (Sign Flag)** - прапорець знаку (співпадає із старшим бітом результату),

**TF (Trap Flag)** - прапорець покрокового режиму (використовується при відладці),

**IF (Interrupt - enable Flag)** - прапорець дозволу апаратних переривань,

**DF (Direction Flag)** - прапорець напряму при строкових операціях,

**OF (Overflow Flag)** - прапорець переповнювання.

**Завдання 3.** Вивчити команди виконання арифметичних операцій над даними з фіксованою комою ADD, ADC, SUB, SBB, MUL, IMUL, DIV, IDIV, CWD, CWB, NEG, INC, DEC. Написати програму, що ілюструє їх використання для заданих констант. Изображение выглядит как снимок экрана, внутренний, монитор

Автоматически созданное описание

Изображение выглядит как снимок экрана, компьютер, внутренний

Автоматически созданное описание

Изображение выглядит как снимок экрана, внутренний, компьютер

Автоматически созданное описаниеИзображение выглядит как снимок экрана, компьютер, внутренний

Автоматически созданное описаниеИзображение выглядит как снимок экрана, внутренний, компьютер, монитор

Автоматически созданное описаниеИзображение выглядит как снимок экрана, внутренний, компьютер, черный

Автоматически созданное описаниеИзображение выглядит как снимок экрана, черный, внутренний, компьютер

Автоматически созданное описаниеИзображение выглядит как снимок экрана, внутренний, черный, монитор

Автоматически созданное описаниеИзображение выглядит как снимок экрана, внутренний, компьютер, черный

Автоматически созданное описаниеИзображение выглядит как снимок экрана, внутренний, монитор, черный

Автоматически созданное описаниеИзображение выглядит как снимок экрана, монитор, внутренний, черный

Автоматически созданное описаниеИзображение выглядит как снимок экрана, черный, внутренний, монитор

Автоматически созданное описаниеИзображение выглядит как снимок экрана, компьютер, монитор, черный

Автоматически созданное описаниеИзображение выглядит как снимок экрана, внутренний, компьютер, черный

Автоматически созданное описание

**Завдання 4.** Представити A/B як десятковий дріб з фіксованою комою. Навести приклад роботи з цим числом.

.define A 223

.define B 101

.define Q A/B ;Q= 2.208

A=0000 0000 1100 1011

B= 0110 0101

DB=0110 0101

O0=2\*A 0 0000 0001 1001 0110 O0>0 O1=2\*O0+DB

+DB 1 0110 0101

O1 1 0110 0110 1110 0110 O1<0 q7=0 O2=2\*O1+B

2\*O1 0 1100 1101 1100 1100

+B 0 0110 0101

O2 1 0011 0010 1100 1100 O2<0 q6=0 O3=2\*O2+B

2\*O2 0 0110 0101 1001 1000

+B 0 0110 0101

O3 0 1100 1010 1001 1000 O3>0 q5=1 O4=2\*O3+B

2\*O3 1 1001 0101 0011 0000

+DB 1 0110 0101

O4 0 1111 1010 0011 0000 O4>0 q4=1 O5=2\*O4+DB

2\*O4 1 1111 0100 0110 0000

+DB 1 0110 0101

O5 1 0101 1011 0110 0000 O5<0 q3=0 O6=2\*O5+DB

2\*O5 0 1011 0110 1100 0000

+B 0 0110 0101

O6 1 0001 1011 1100 0000 O6<0 q2=0 O7=2\*O6+B

2\*O6 0 0011 0111 1000 0000

+B 0 0000 1001

O7 0 0100 0000 1000 0000 O7>0 q1=1 O8=2\*O7+DB

2\*O7 0 1000 0001 0000 0000

+DB 1 0110 0101

O8 1 1110 0110 0000 0000 O8<0 q0=0 O=O8+B

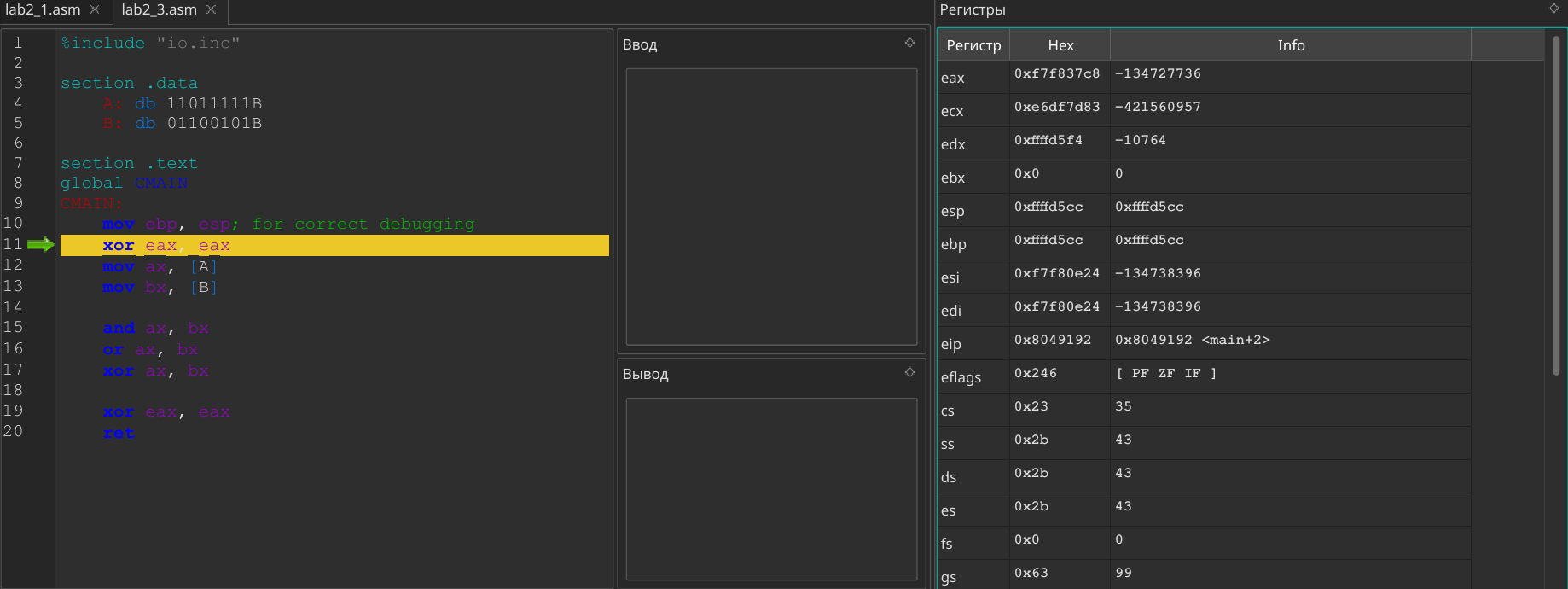
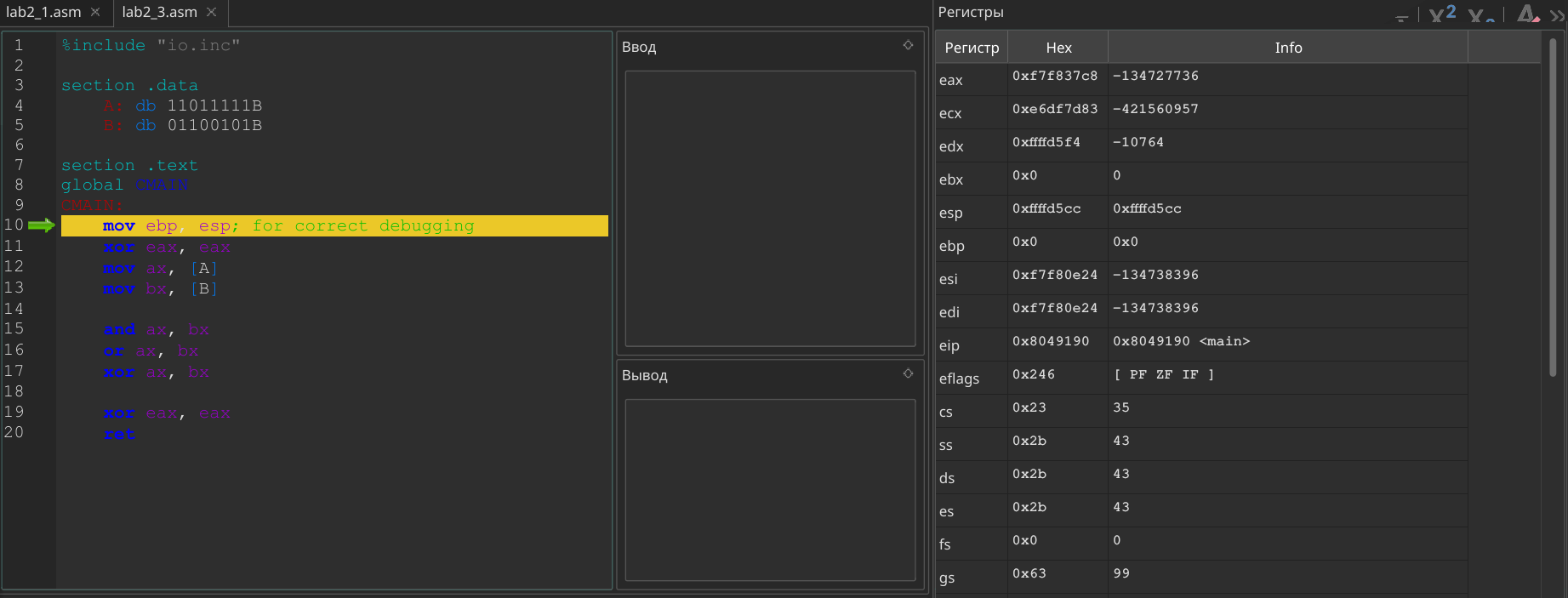
+B 0 0000 1001

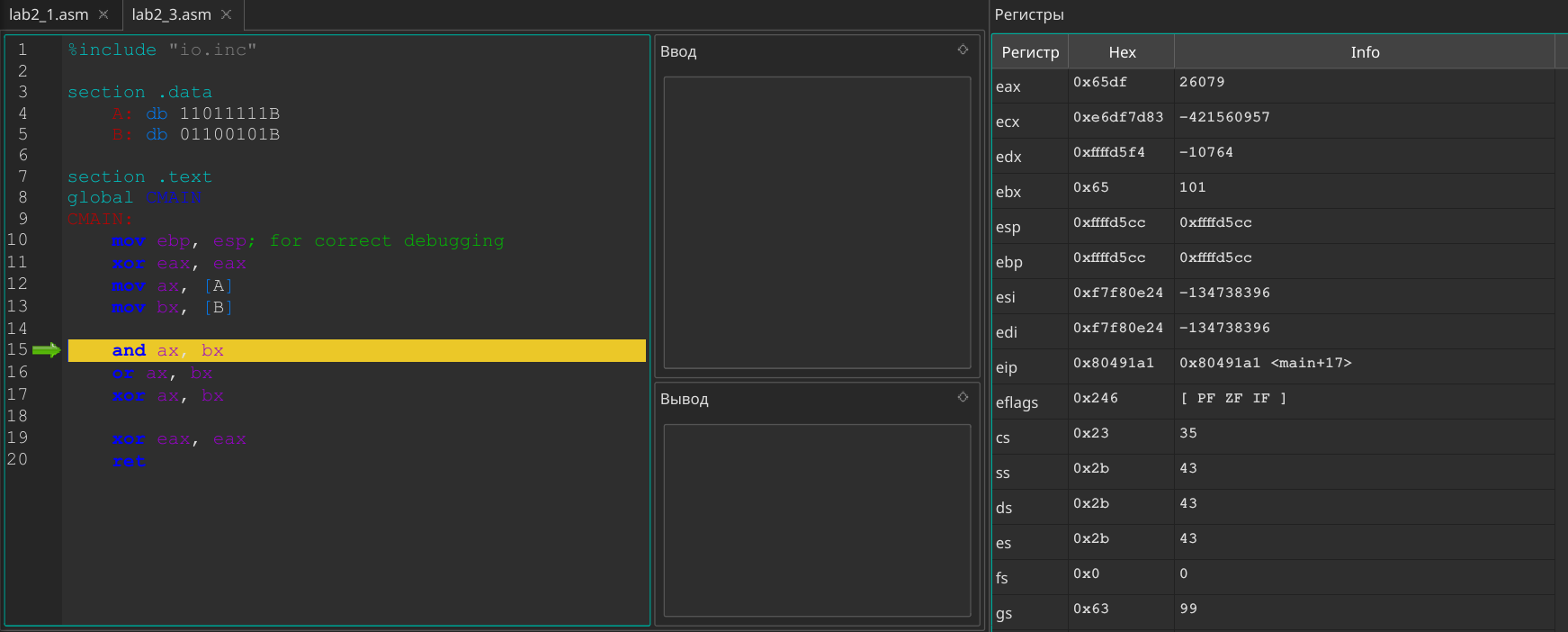
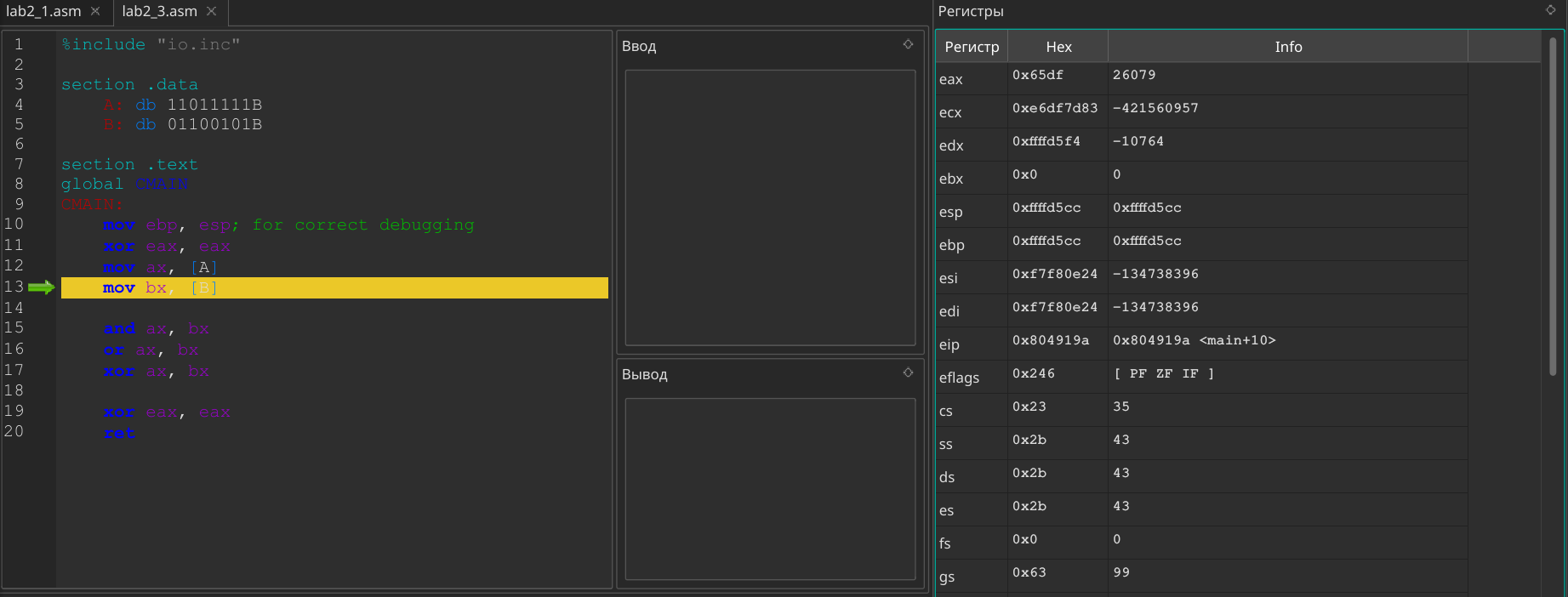
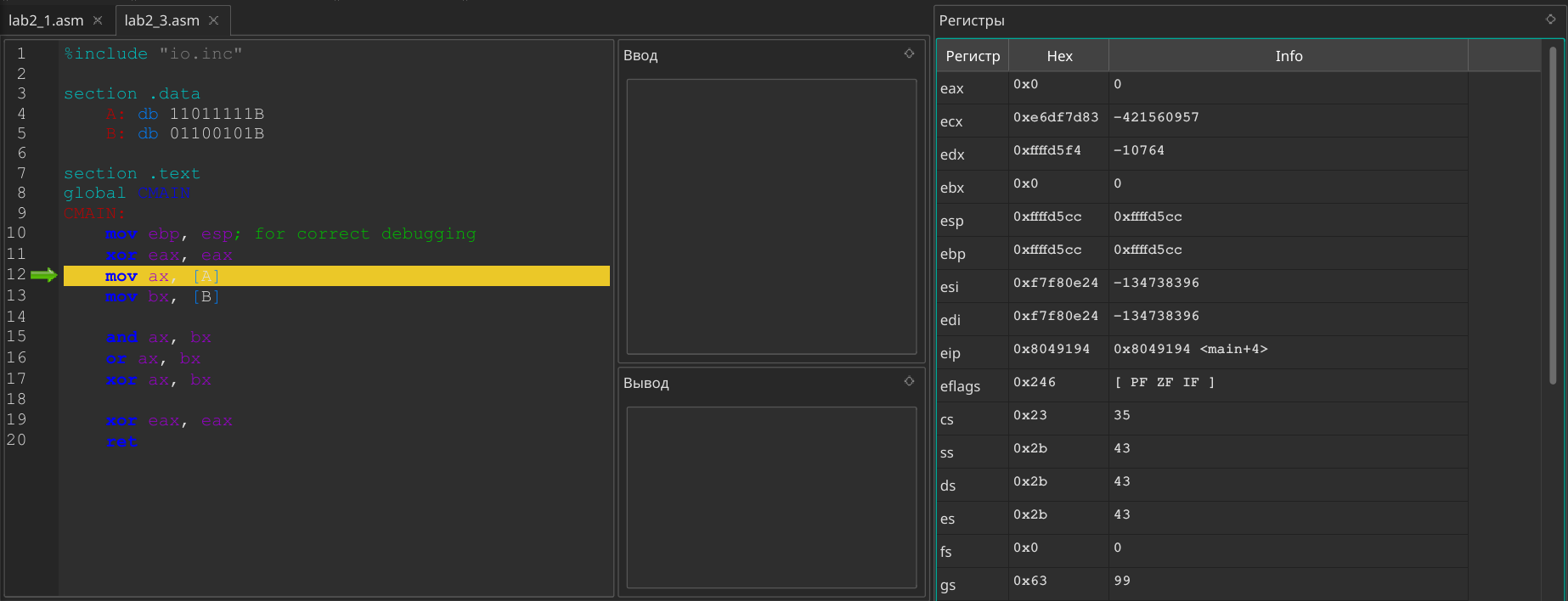
O 1 1110 1111 0000 0000

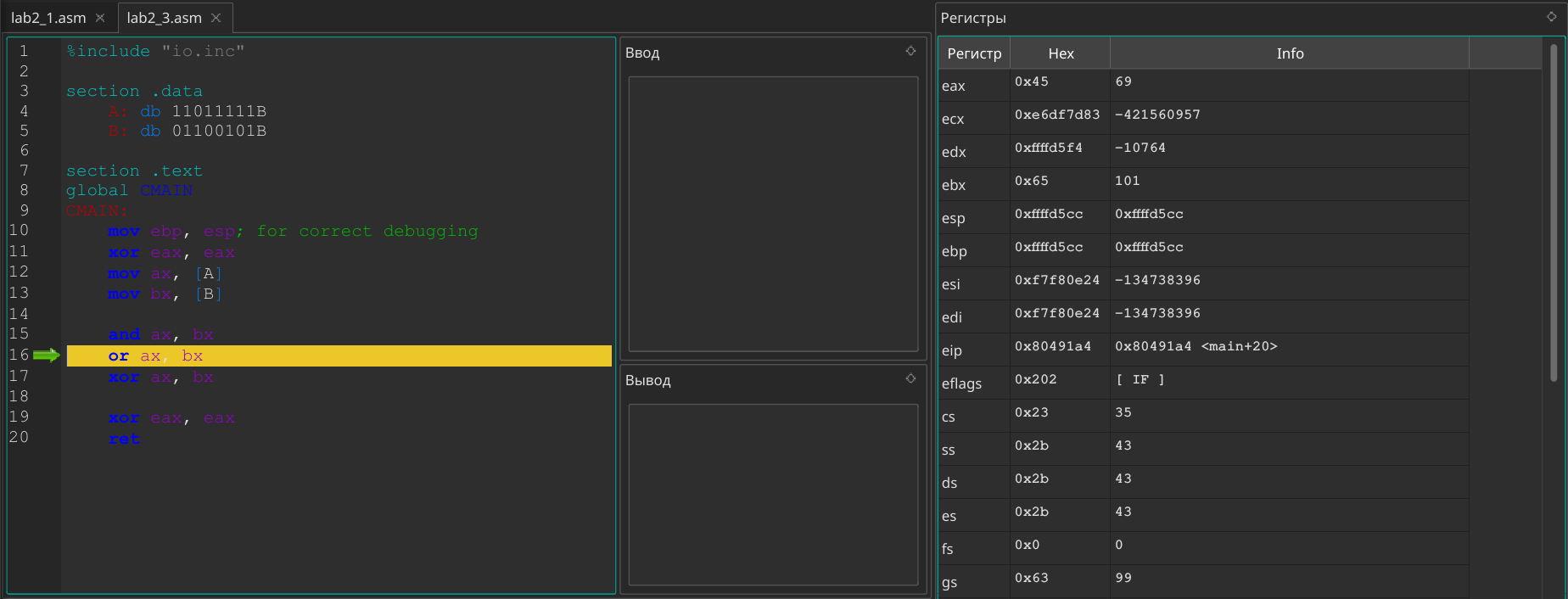
Q=q7q6q5q4q3q2q1q0=0110 0010=50

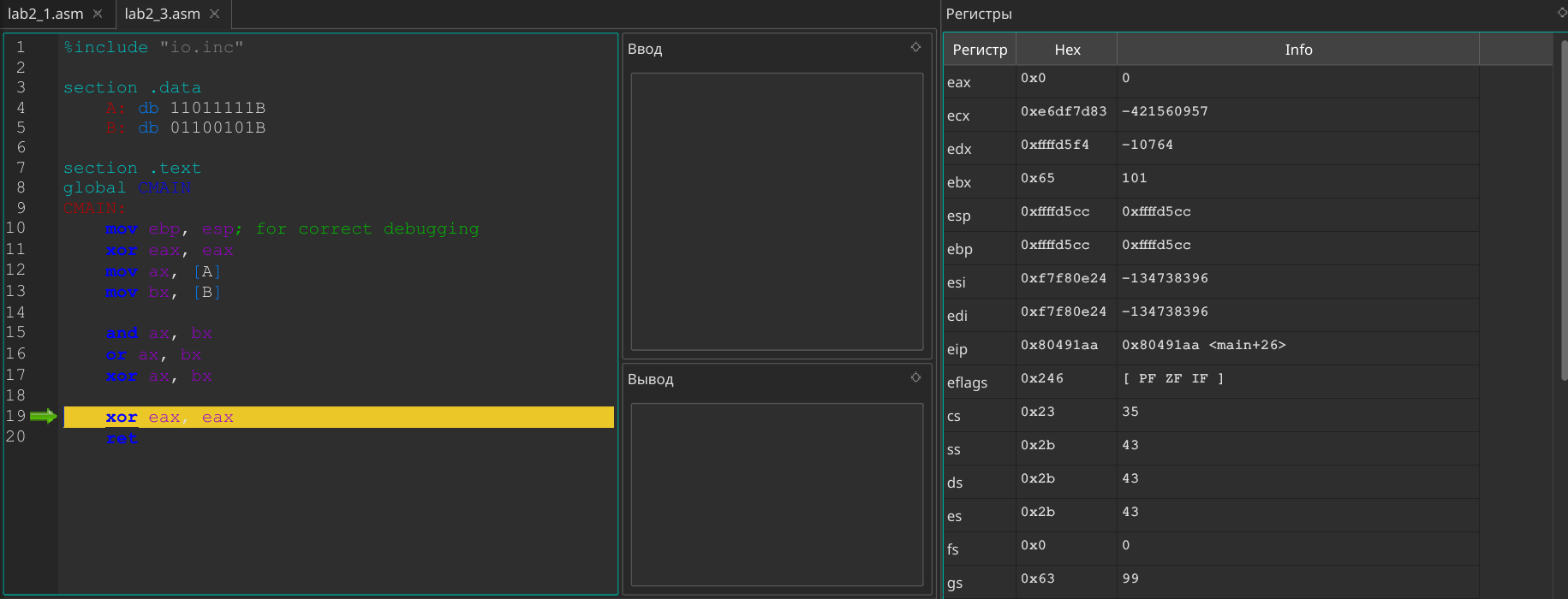
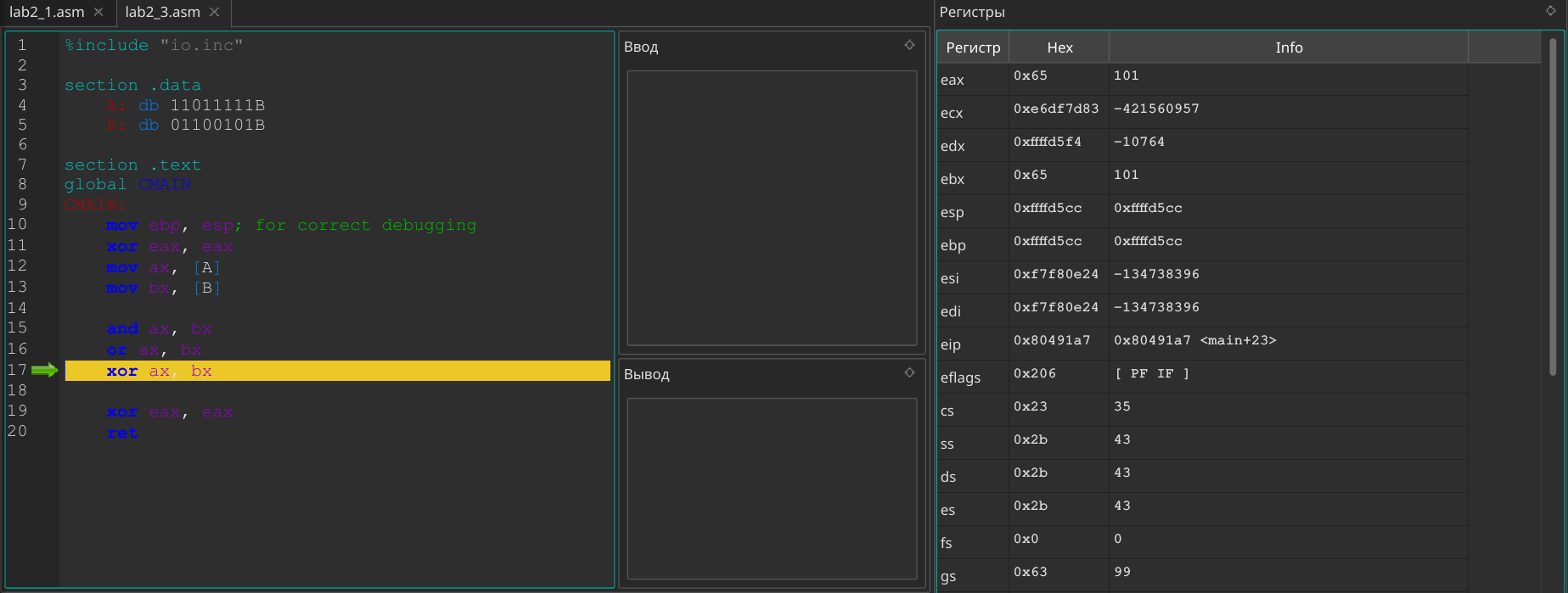
O= 1110 1111

**Завдання 5.** Вивчити команди виконання логічних операцій AND, OR, XOR, продемонструвати їх роботу за допомогою демонстраційної програми.

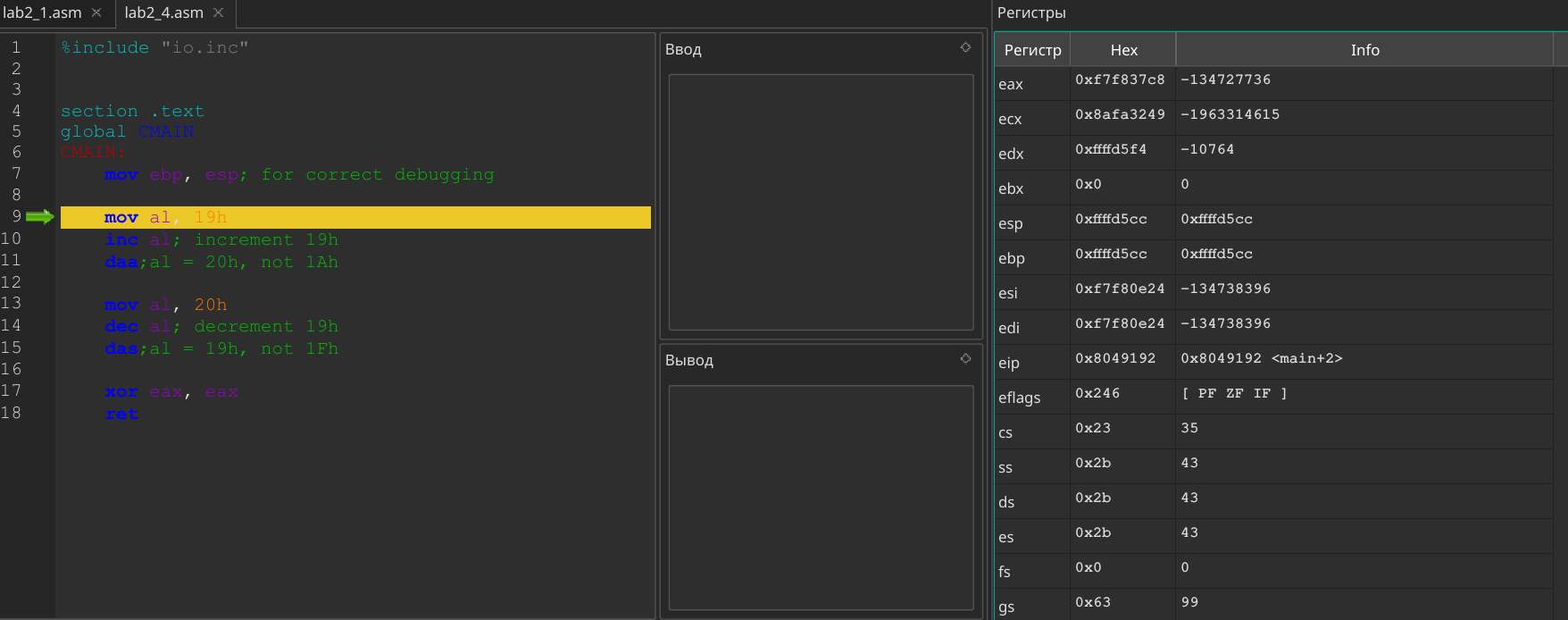
****

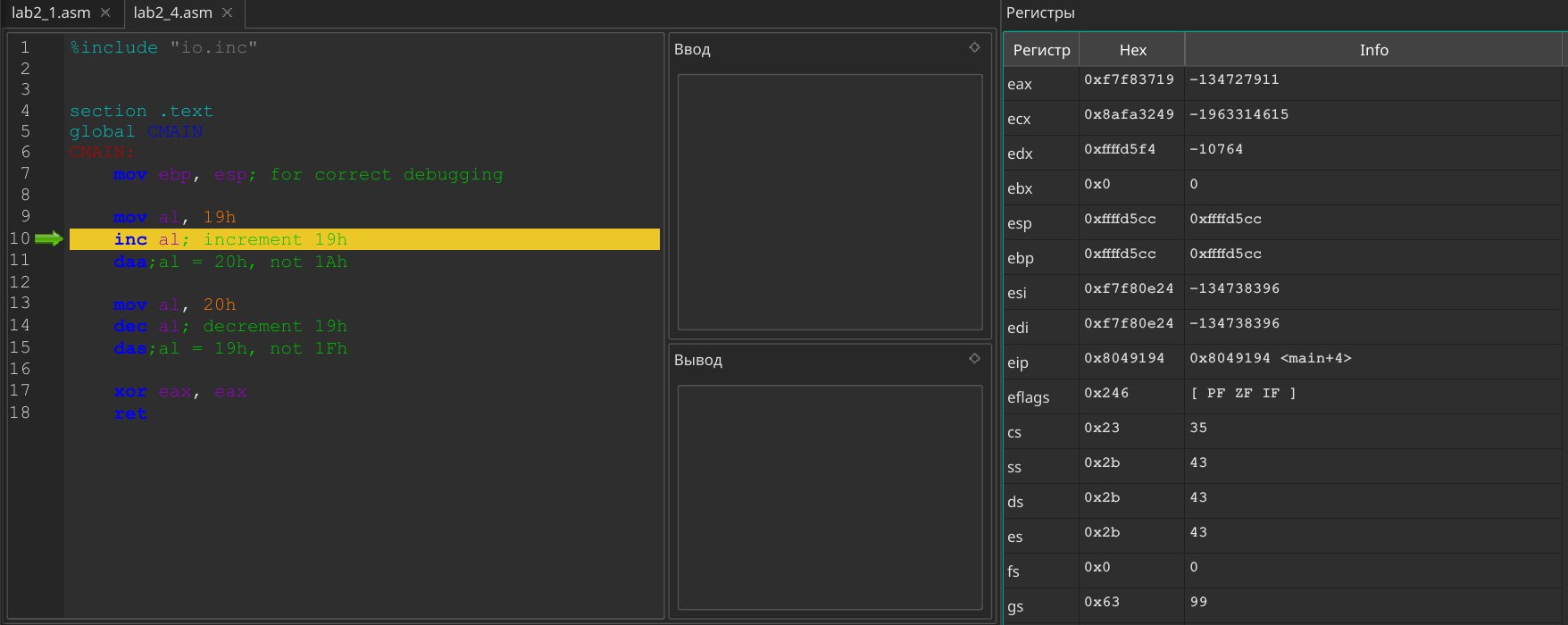


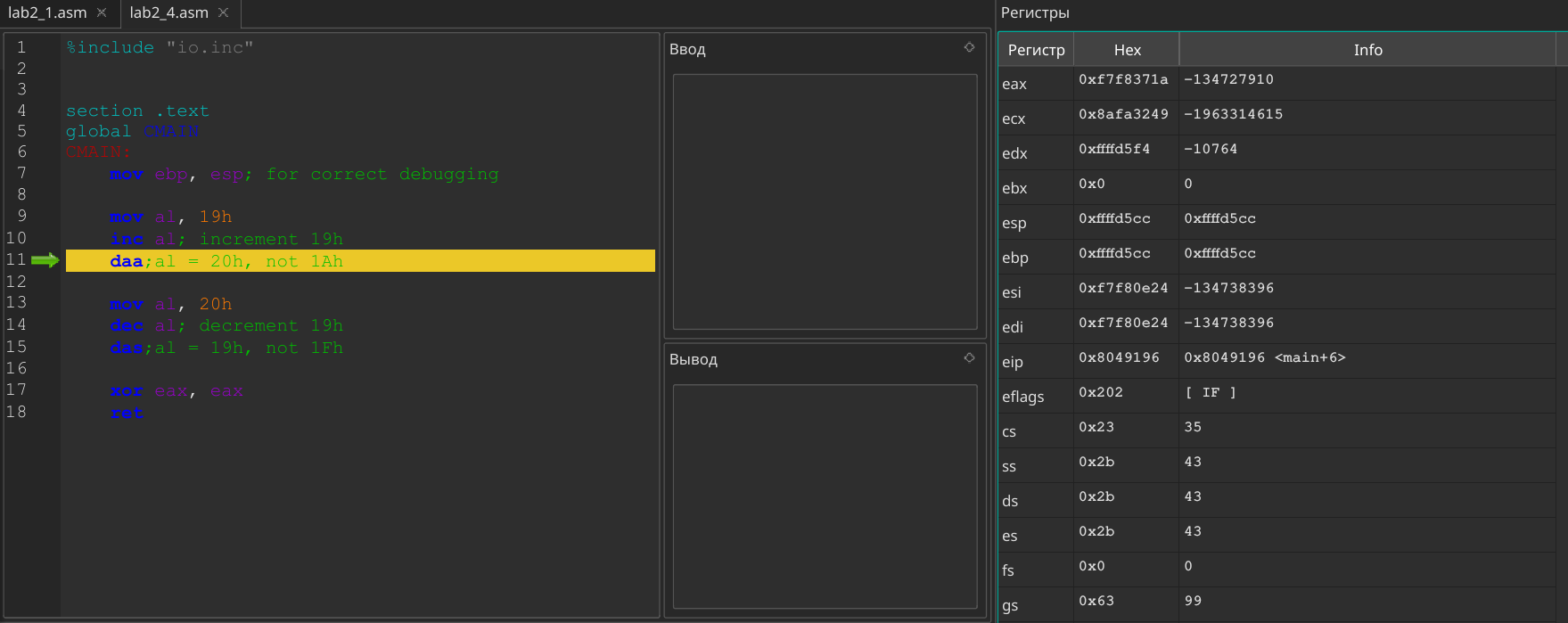


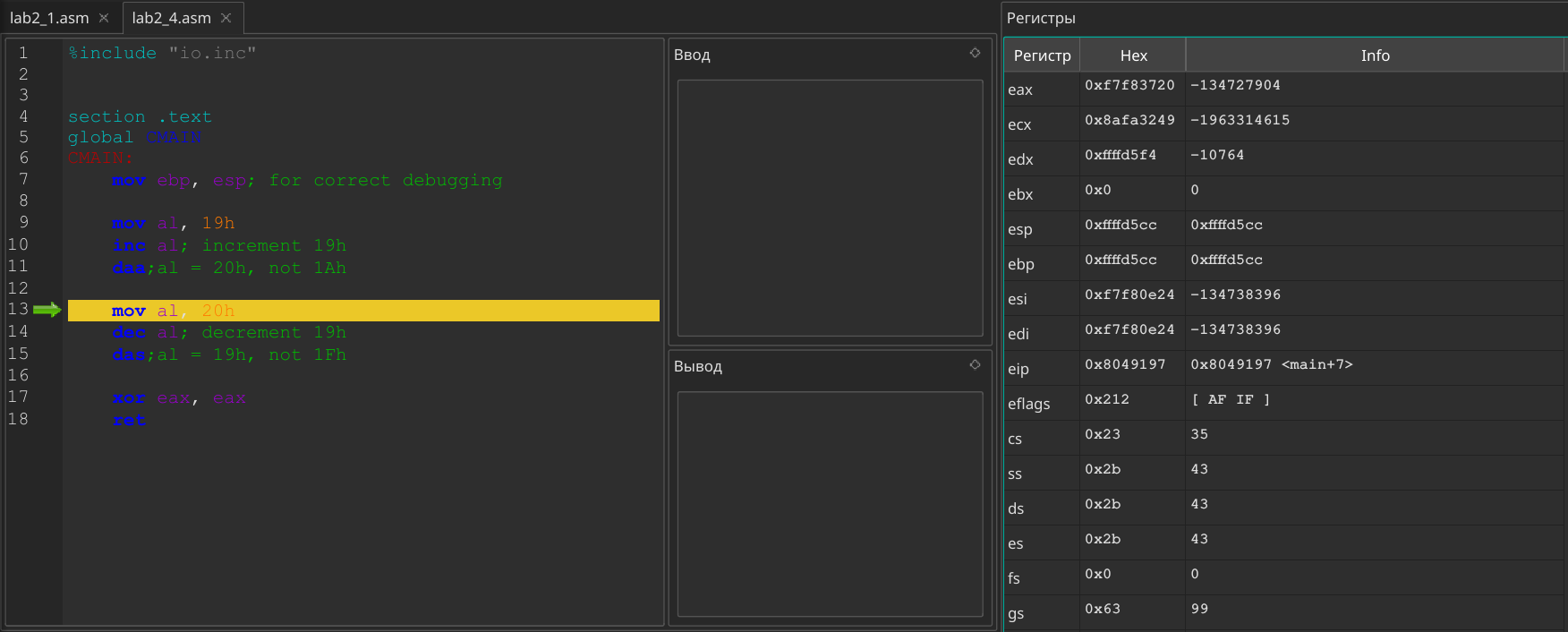


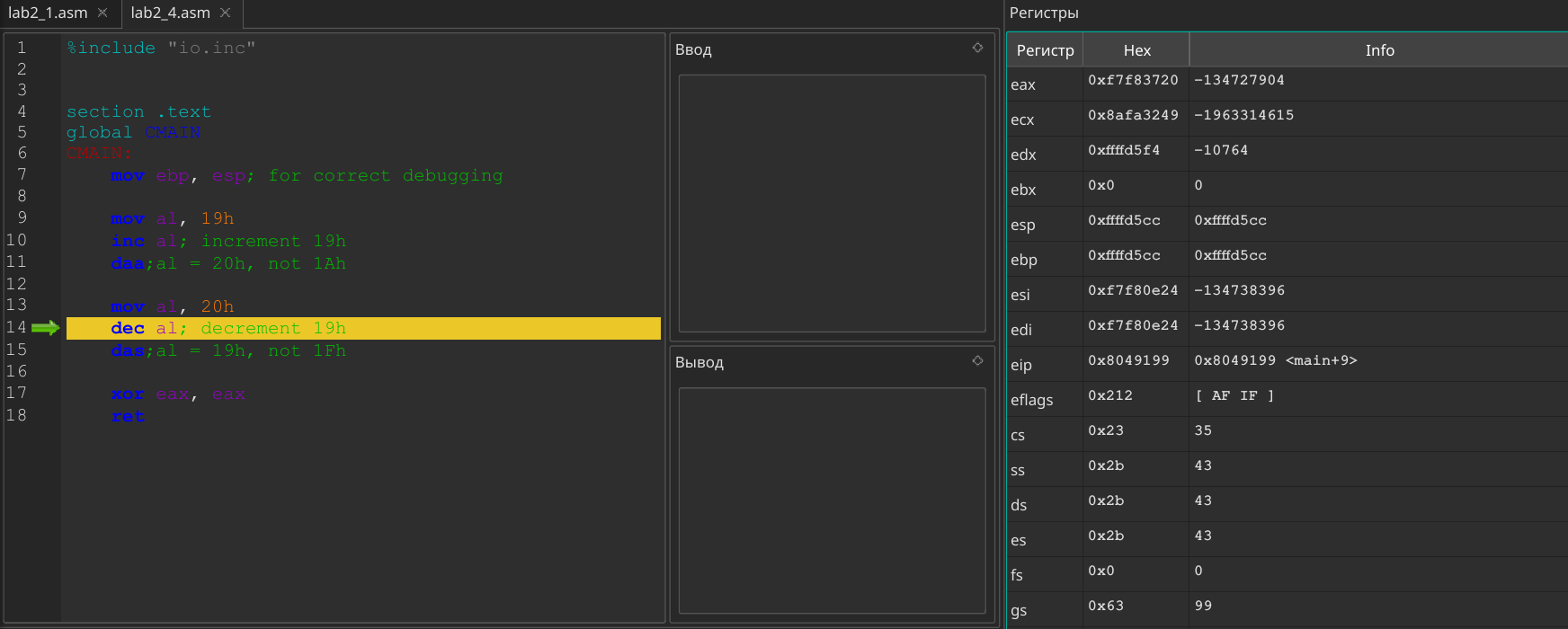
**Завдання 6**. Вивчити команди виконання операцій над двійково-десятковими числами в упакованому форматі. Навести приклади використання команд DAA і DAS.

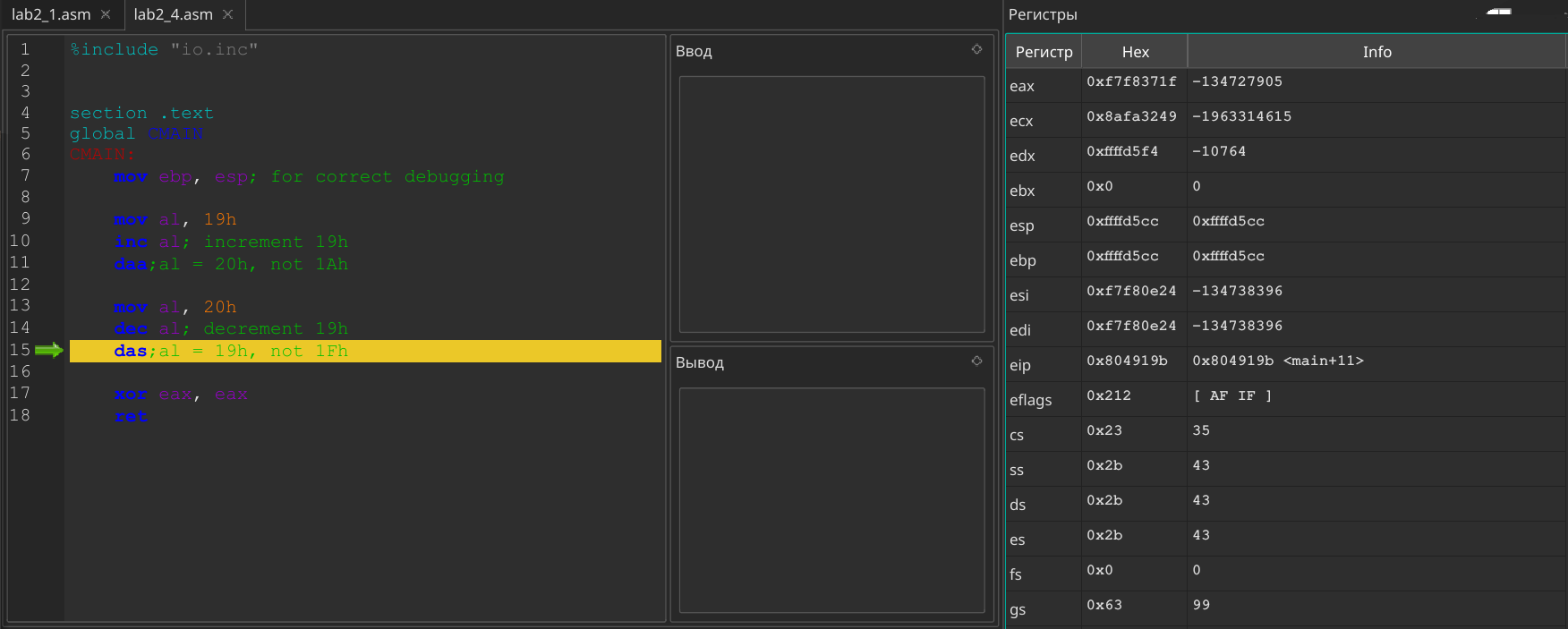


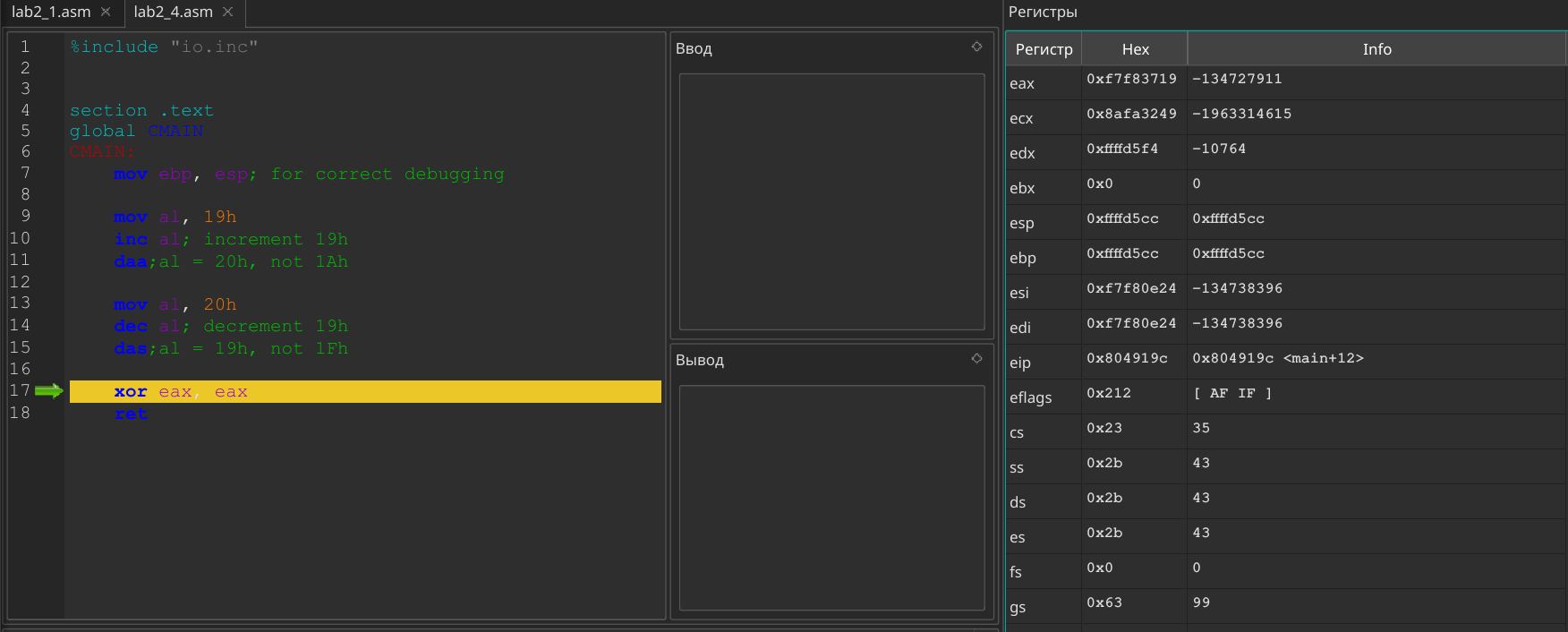
****

****

****

****

****

****

**Висновок:** у ході другої лабораторної роботи було досліджено роботу АЛУ, його регістри. Досліджено різноманітні системи числення, що широко

застосовуються у обчислювальній техніці: двійкова система з фіксованою крапкою, шістнадцятирічна, BCD – упакована двійково-десяткова система. Вивчено основи синтаксису Assembler та основні команди для виконання елементарних операцій над числами у різних форматах.