

Практическое занятие №3

Тема: Составление программ ветвящейся структуры в IDE PyCharm Community.

Цель: Закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ ветвящейся структуры в IDE PyCharm Community.

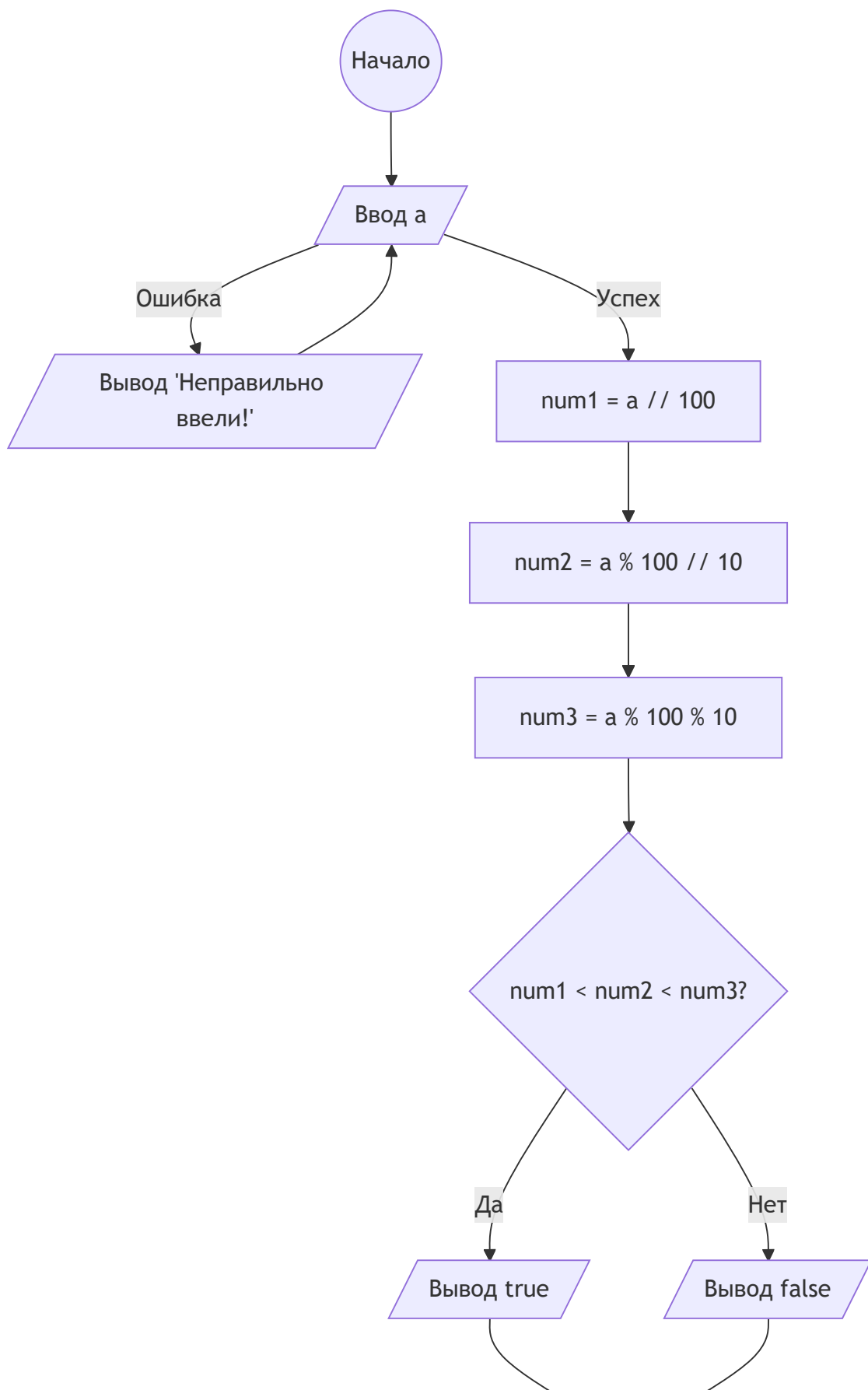
Задание №1

Постановка задачи:

Дано трехзначное число. Проверить истинность высказывания: «Цифры данного числа образуют возрастающую последовательность».

Тип алгоритма: с ветвлением.

Блок-схема алгоритма:





Текст программы:

```
try:
    a = int(input("Введите целое трехзначное число: "))

    num1 = a // 100
    num2 = a % 100 // 10
    num3 = a % 100 % 10

    if num1 < num2 < num3:
        print("true")
    else:
        print("false")
except ValueError:
    print("Неправильно ввели!")
```

Протокол работы программы (примеры):

```
Введите целое трехзначное число: 123
true

Введите целое трехзначное число: 321
false

Введите целое трехзначное число: 121
false

Введите целое трехзначное число: abc
Неправильно ввели!
Введите трехзначное число: 123
true
```

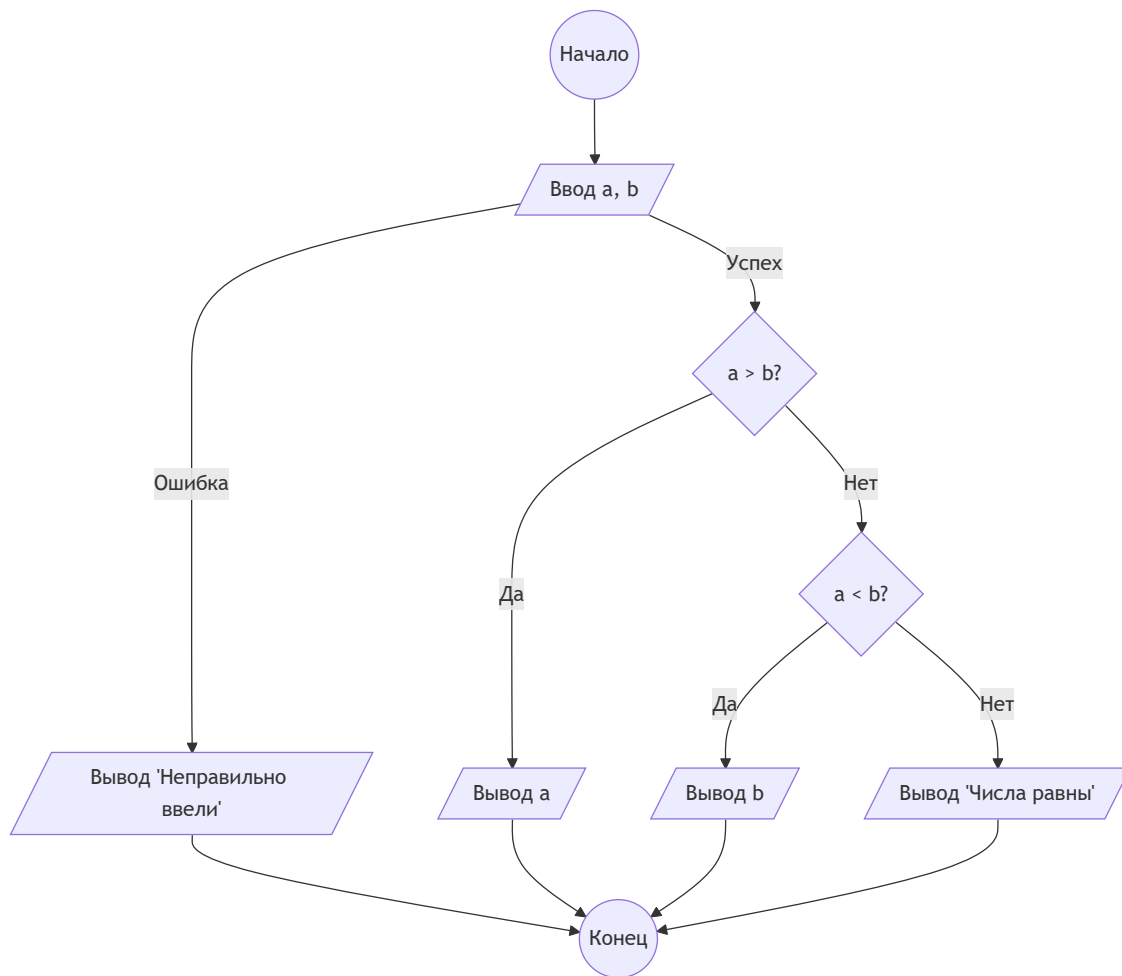
Задание №2

Постановка задачи:

Даны два числа. Вывести большее из них. Если числа равны, вывести сообщение об этом.

Тип алгоритма: с ветвлением.

Блок-схема алгоритма:



Текст программы:

```

try:
    a = int(input('Введите первое число: '))
    b = int(input('Введите второе число: '))
except ValueError:
    print("Неправильно ввели")
    exit() # Завершаем программу при ошибке ввода

if a > b:
    print(a)
elif a < b: # Используем elif для большей ясности
    print(b)
else:
    print("Числа равны")
  
```

Протокол работы программы (примеры):

```

Введите первое число: 5
Введите второе число: 3
5
  
```

```
Введите первое число: 1
Введите второе число: 7
7
```

```
Введите первое число: 4
Введите второе число: 4
Числа равны
```

```
Введите первое число: a
Неправильно ввели
```

Вывод:

В ходе выполнения практического задания были закреплены навыки использования условного оператора, операций деления нацело (`//`) и взятия остатка от деления (`%`) для работы с цифрами числа, навыки использования условного оператора для сравнения чисел. Программа корректно определяет большее из двух чисел или выводит сообщение о равенстве. Блок-схема отражает логику программы. Код немного улучшен за счет использования `elif` и завершения программы при ошибке ввода, чтобы избежать дальнейших ошибок.. Программа корректно проверяет, образуют ли цифры трехзначного числа возрастающую последовательность, и обрабатывает ошибки ввода. Блок-схема отражает логику программы.