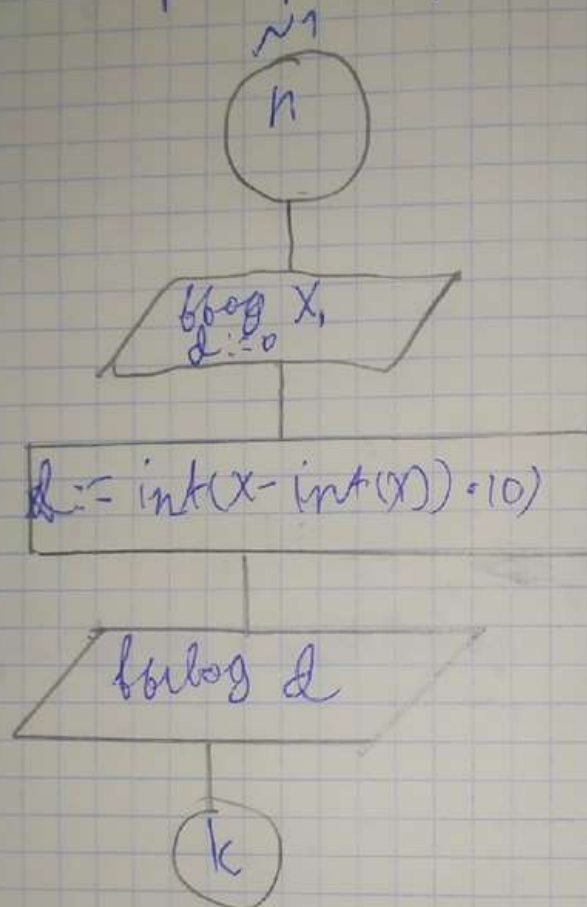
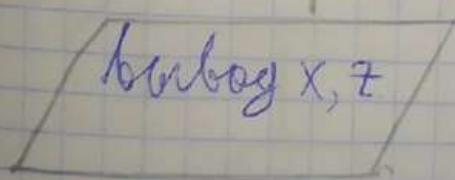
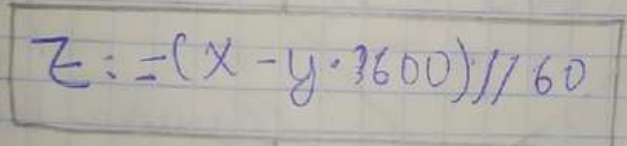
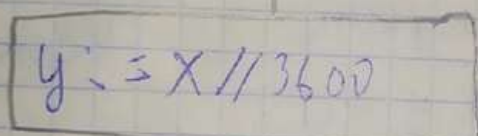
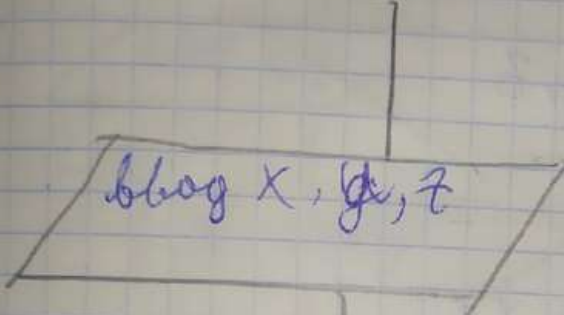
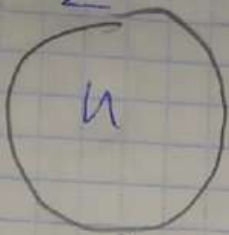


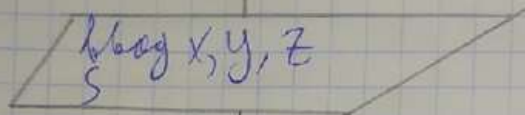
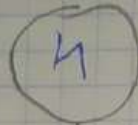
лабораторная работа номер 1



$\sim 2$



~3



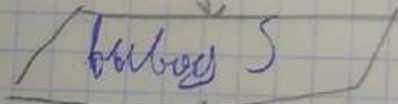
if  $x \geq 0$   
and  $x \leq 11$   
and  $y \geq 0$   
and  $z \leq 89$

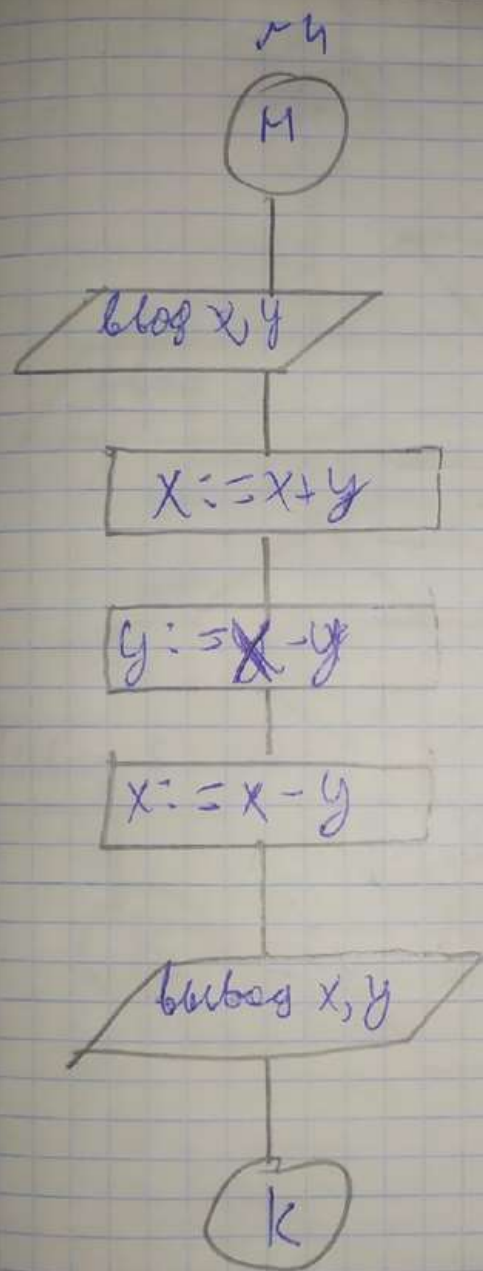
Ya

Ken

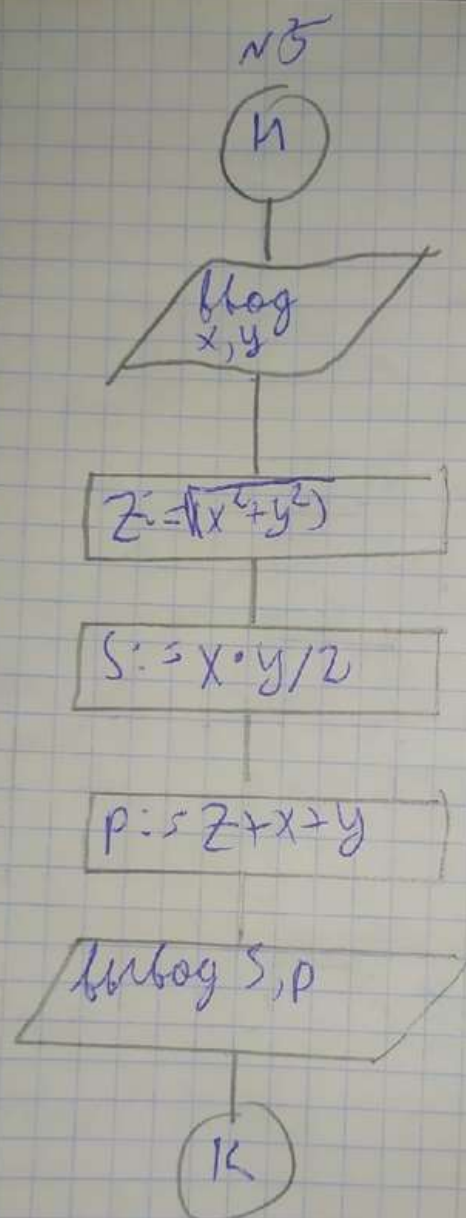
Output Hoga

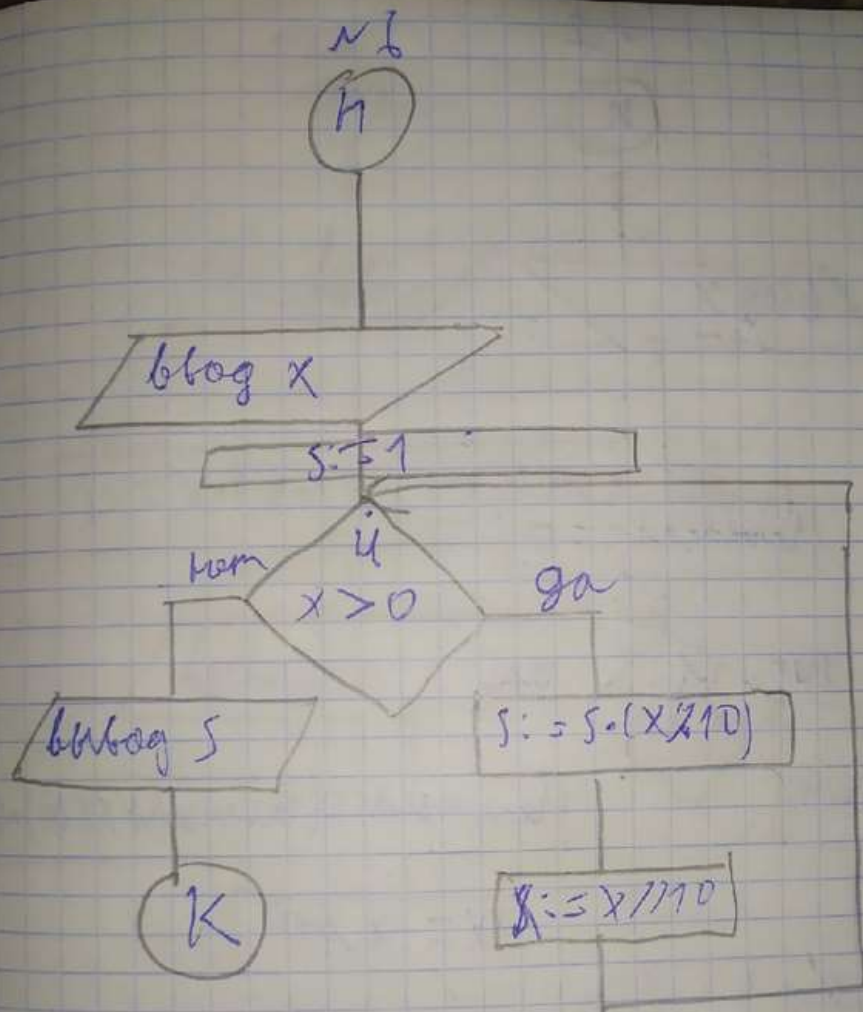
$$S := 12 \cdot x + y/2 + z \left( \frac{y^5}{85} \right)$$

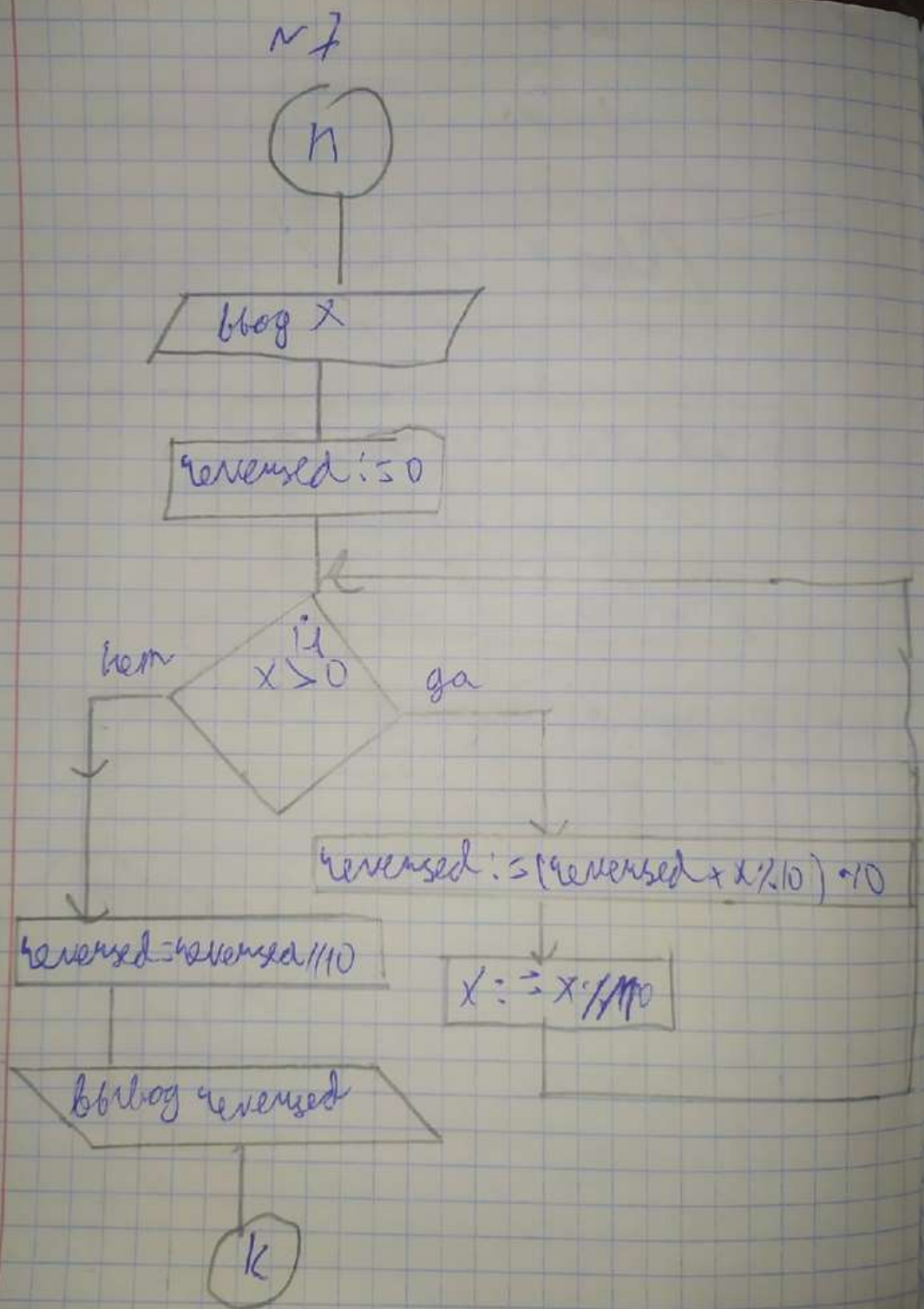












$\sim 8$

(n)

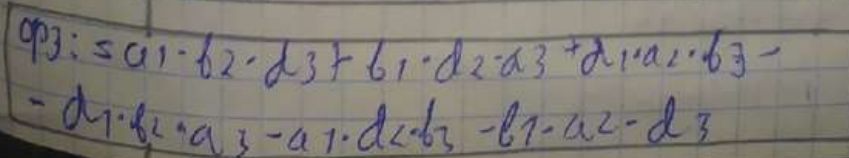
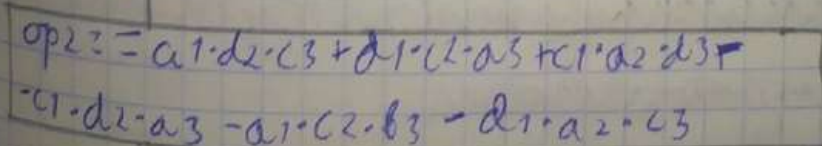
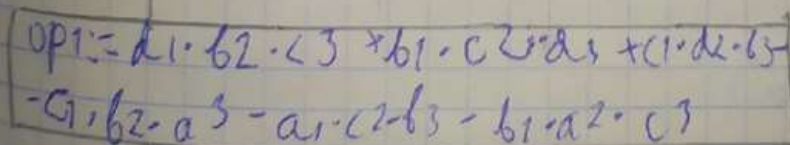
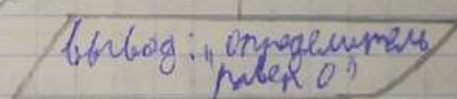
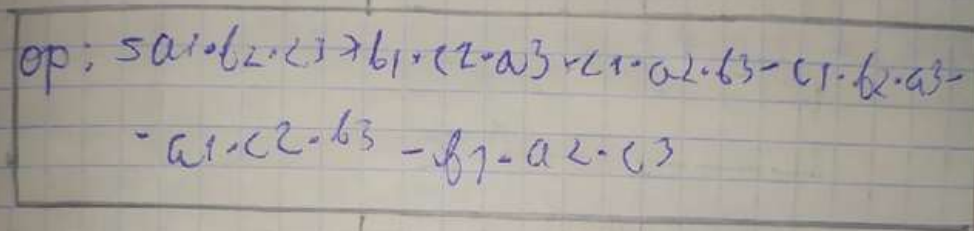
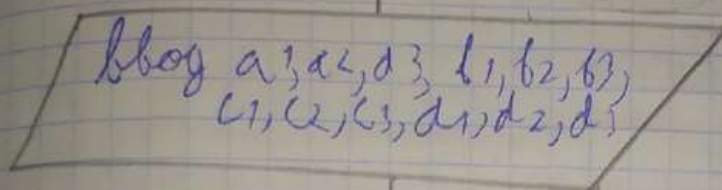
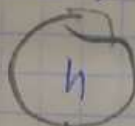
$\log X$

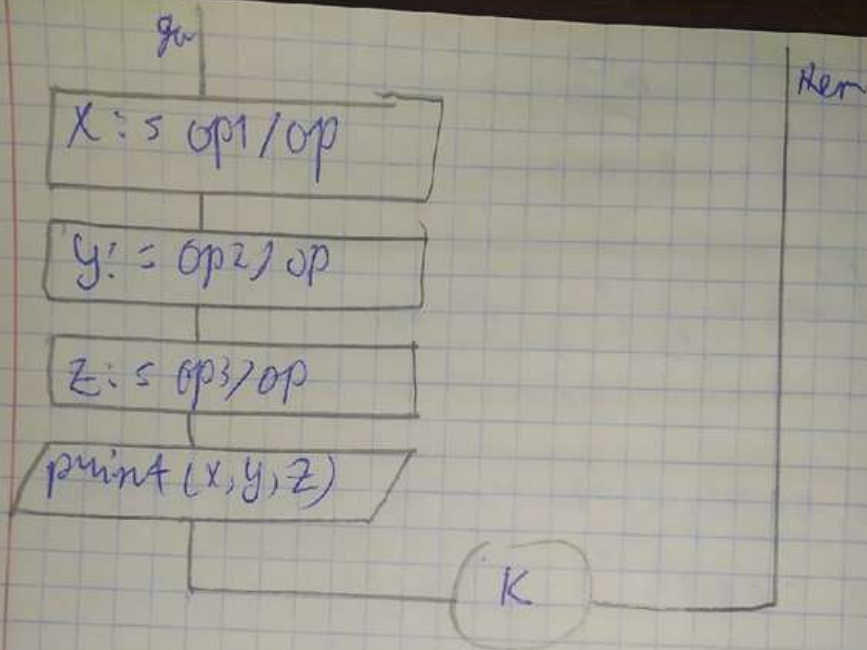
$$S \leftarrow X \cdot (X + (X \cdot (3 \cdot X - 5) + 2) - 7) \cdot 2$$

$\log S$



N9





### Контрольные вопросы

1) Алгоритм — конечная последовательность однозначных действий, приводящая к решению поставленной задачи.

Свойства:

- детерминированность (определенность)
- результативность
- массовость
- дискретность



Виды:

- Блок-схема
- линейный
- ветвистый

2) Компиляция - преобразование исходного кода программы в машинный код.

Компилятор - включение нужных функций из сторонних библиотек, которые используются в программе.

3) 1) Интерпретивное программирование:

- В исходном коде программы записываются инструкции
- Инструкции должны выполняться последовательно.

- данные, полученные при выполнении инструкций могут записываться в память.

2) Декларативное программирование:

- задаётся спецификация реше-  
ние задачи (описывается проблема  
и ожидаемый результат)

3) Структурное программирование:

- представление программы в  
виде иерархической структуры блоков

4) Функциональное программирование:

- процесс вычисления трактуется  
как вычисление значений математических функций

5) Логическое программирование:

- абстрактное дост-во теорем

- раздел дискретной математики,  
изучающий принципы логического  
ввода информации

6) Объектно-ориентированное программирование:

- программа в виде совокупности  
объектов.



4) Создание консольного приложения на C#:

1) Запускаем Visual Studio

2) На начальном экране выбираем "Создать проект"

3) В появившемся окне "Создание проекта" выбираем консоль. Затем выбираем C# в качестве языка и Windows в качестве платформы.

4) Выбираем тип консольного приложения (.NET Core)

Ctrl + F5 - запускаем отладку

Shift + F6 - прерываем отладку

5) При помощи клавиш F5 запускаем программу. При возникновении ошибок или предупреждений в строку

6) среда C# - механизм, управляющий

иной при необходимости.

Библиотека `libc` - позволяет программистам использовать стандартные, стандартные код.

7) `char` - (маленький) (8 бит)

- `int` - целый (32 бита)

- `float` - вещественный (4 бита)

- `double` - вещественный двойной точности (64 бита)

- `void` - не имеет значения (0 бит)

8) Переменная - математический объект, который занимает некоторое множество значений.

Константы не изменяются во время выполнения программы. Объявляются модификатором `const`. Переменные объявляются `foo int` использование

9) Инициализация `var` позволяет не повторять имя типа при



объявлении переменных.

10) после объявления переменной её нельзя объявить повторно или указать её значение другого типа.

В C# есть следующие способы преобразования:

- явное (преобразование (помощью оператора приведения типов))
- неявное (осуществляется автоматически)
- пользовательское (при помощи операторов пользовательского преобразования)

11) префиксный элемент ставится перед переменной, а постфиксный - после

12) при выводе строк на экран используется метод `Console.WriteLine()` или `Console.WriteLine()`.