



ИТ-КЛАСТЕР  
СИБИРИ

# Открытая всероссийская конференция **Преподавание информационных технологий в РФ**

## **Элементы продуктового подхода при обучении программированию в школе**

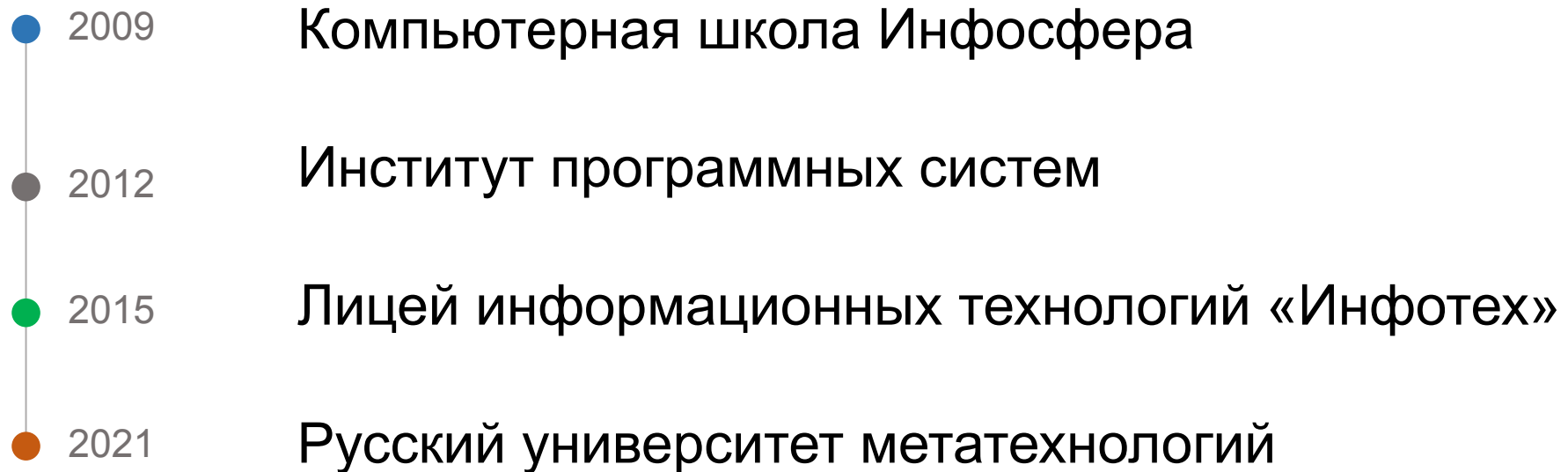
Гарипова Ю.Е., Лицей «Инфотех»  
Жеребцова О.В., институт iSpring

15-16 мая 2025 года, г. Омск

# Образовательная система iSpring



Глобальная ИТ-компания из Йошкар-Олы, разработчик платформы для корпоративного онлайн-обучения



# Докладчики



Жеребцова Ольга Вениаминовна,

руководитель административного отдела института iSpring,  
старший преподаватель кафедры программных систем института iSpring,  
преподаватель информатики и программирования лицея «Инфотех»,  
член жюри олимпиад по информатике и программированию



Гарипова Юлия Евгеньевна,

кандидат технических наук, доцент кафедры проектирования и производства  
электронно-вычислительных систем ФГБОУ ВО ПГТУ,  
преподаватель программирования лицея «Инфотех»,  
преподаватель компьютерной школы «Инфосфера»

# Что важно для индустриального программирования

- 1) продумать алгоритмы и структуры данных;
- 2) выбрать оптимальные алгоритмы для решаемой задачи с точки зрения эффективности по времени;
- 3) обеспечить работоспособность продукта на любых вариантах входных данных;
- 4) обеспечить удобство пользования продуктом;
- 5) проектировать продуктивное решение, удобное для дальнейшей совместной разработки:
  - обеспечить поддерживаемость программного кода путём форматирования, выделения независимых программных единиц с учётом их безопасности;
  - продумать эффективность с точки зрения времени разработки и масштабируемости программного решения.

# Что заложено в требованиях ФГОС

- 1) продумать алгоритмы и структуры данных;
- 2) выбрать оптимальные алгоритмы для решаемой задачи с точки зрения эффективности по времени;
- 3) обеспечить работоспособность продукта на любых вариантах входных данных;
- 4) обеспечить удобство пользования продуктом;
- 5) проектировать продуктивное решение, удобное для дальнейшей совместной разработки и поддержки:
  - обеспечить поддерживаемость программного кода путём форматирования, выделения независимых программных единиц с учётом их безопасности;
  - продумать эффективность с точки зрения времени разработки и масштабируемости программного решения.

# Преподавание в лицее: акцент на разработку **продукта**

- формировать набор тестовых данных, согласовывать формат и вид выходных данных с заказчиком (учителем)
- обеспечить поддерживаемость программного продукта путём:
  - форматирования и именования переменных;
  - разделения кода на подзадачи:  
ввод данных, решение задачи / расчёт результата, вывод ответа;
- анализировать и корректировать решение с точки зрения удобства поддержки и масштабируемости;
- тестировать решение поэтапно в процессе разработки (юнит-тестирование).

# Практика преподавания в лицее: 7-8 класс

При изучении основных алгоритмических конструкций уже можно формировать:

- навык оформления кода по принятым правилам: именование идентификаторов в соответствии со смыслом, структурирование кода с учетом сдвигов, пробелов...
- привычку формировать набор тестовых данных для структурного тестирования продукта,
- подход к проектированию / анализу решения с точки зрения совместной разработки и масштабируемости.

# Практика преподавания в лицее: 7-8 класс

**Цель:** подготовить учеников к их будущей жизни (деятельности) так, чтобы они были конкурентоспособны

- Делаем акцент на мышление, а не на язык
- Подбираем задачи так, чтобы они требовали проработки
  - тестовых данных,
  - подхода к решению
- Проверяем не только и не столько работоспособность, сколько алгоритм и его оформление (форматирование).

**Следствие:** конкретные инструменты языка второстепенны, важнее привить культуру программирования – хорошо поставленная база позволяет быстро освоить необходимые технологические средства любого языка программирования



# Практика преподавания в лицее: проверка задач

```
//6
PROGRAM line(INPUT, OUTPUT); //форматирование
VAR
  A, B, Num: INTEGER;
  Segment: BOOLEAN;
BEGIN
  WRITELN('Введите границы числового отрезка ');
  READLN(A);
  READLN(B);
  WRITELN('Введите число ');
  READLN(Num);
  IF Num >= A
  THEN
    IF Num <= B
    THEN
      Segment := TRUE
      //ELSE ....не входит в числовой отрезок, отметить!
    ELSE
      Segment := FALSE;
  IF Segment = TRUE //TRUE не пишем, поэтому имя взять InSegment, будет точнее
  THEN
    WRITELN(Num, ' входит в числовой отрезок от ', A, ' до ', B)
  ELSE
    WRITELN(Num, ' не входит в числовой отрезок от ', A, ' до ', B);
```

# Практика преподавания в лицее: проверка задач

```
//9
PROGRAM BelongsToTheCut(INPUT, OUTPUT);
VAR
  A, B, Num: INTEGER;
  Answer: BgOOLEAN; //плохое имя, ни о чем не говорит!
BEGIN
  WRITE('Введите границу A: ');
  READLN(A);
  WRITE('Введите границу B: ');
  READLN(B);
  WRITE('Введите число: ');
  READLN(Num);
  //можно было изначально взять Answer := FALSE, тогда код красивее
  IF Num >= A
  THEN
    IF Num <= B
    THEN
      Answer := TRUE
    ELSE
      Answer := FALSE
  ELSE
    Answer := FALSE;
  IF Answer
  THEN
    WRITELN('Число ', Num, ' принадлежит отрезку [' , A, ', ', B, ']')
```

# Практика преподавания в лицее: проверка задач

```
//7
PROGRAM FindMaxNumber(INPUT, OUTPUT);
VAR
    Num, FirstNumber, SecondNumber, FirstMore: INTEGER;
BEGIN
    WRITE('Введите двухзначное число: ');
    READLN(Num);
    FirstNumber := Num DIV 10;
    SecondNumber := Num MOD 10;
    //начальное значение FirstMore?
    IF FirstNumber > SecondNumber
    THEN
        FirstMore := 1; //чему равно FirstMore, если не выполнится?
    IF FirstMore = 1
    THEN
        WRITE('Наибольшее число равно', FirstNumber, SecondNumber) //форматирование
    ELSE
        WRITE('Наибольшее число равно: ', SecondNumber, FirstNumber); //форматирование
    //на экране появилась надпись, а число так и не получено
    READLN
END.
```

# Практика преподавания в лицее: сборник задач

## 2. Задачи на тему «Ветвление»

2.1. Вводится целое число  $X$  ( $-10\,000 \leq X \leq 10\,000$ ). Для введенного числа определить и сообщить:

- a) является ли оно положительным или отрицательным;
- b) является ли оно четным;
- c) является это число двухзначным;
- d) делится ли это число на свою последнюю цифру.

2.2. Вводится двухзначное положительное число. Сообщить какая из его цифр больше – десятки или единицы.

2.3. Вводится два целых числа, не превышающих по абсолютному значению 30 000. Для этих чисел:

- a) определить и вывести максимальное значение;
- b) вывести их в порядке возрастания;
- c) определить максимальное четное число или сообщить, что таких нет;
- d) сообщить, являются ли оба числа с одним знаком (плюс или минус, считать ноль одним знаком с положительным числом);
- e) сообщить, делится ли одно число на другое.

2.4. Вводится два целых числа (от -30 000 до 30 000). Сообщить какое число больше другого и на сколько? Может, они равны? Например, для чисел 6 и 8 должен быть дан ответ «второе больше первого на 2».

2.5. Вводятся границы отрезка  $[A; B]$ , а потом число. Сообщить, принадлежит ли число указанному отрезку. Ответ дать словами. Все вводимые числа целые, от -30 000 до 30 000.

2.6. Вводится положительное двухзначное число. Нужно переставить его цифры так, чтобы получить максимально возможное число. Получить и вывести новое число.

9

2.7. Вводится положительное трехзначное число  $X$ . Из него можно вычеркнуть одну цифру так, чтобы оставшееся число  $Y$  являлось делителем исходного числа  $X$ . Чему равно  $Y$ ?

2.8. Найти решение линейного уравнения в формате  $Ax + B = 0$ , где вводятся коэффициенты  $A$  и  $B$  – целые числа от -30 000 до 30 000. Ответ округлить до сотых.

2.9. Вводится два числа: цена монитора SamsungSyncMaster в январе 2018 г. и январе 2019 г. Оба числа целые и не превышают 30 000. Произошло удешевление или подорожание? На сколько процентов изменилась цена? В ответе число процентов округлить до целого.

2.10. Маша и Миша собирали землянику. Маше удалось сорвать  $X$  ягод, а Мише –  $Y$  ягод. Поскольку ягода была очень вкусной, то ребята могли какую-то часть ягод съесть. По нашим подсчетам вместе они съели  $Z$  ягод. Требуется определить: сколько ягод ребята принесли домой в результате, при этом следует проверить, не ошиблись ли мы в расчетах, подсчитывая количество съеденных ягод (их не должно было получиться больше, чем сорванных ягод). Все числа целые неотрицательные, не превышают 30 000.

2.11. В магазине покупают книгу. Программа запрашивает стоимость книги и сумму денег, внесенную покупателем. Если сдачи не требуется, печатает «Спасибо». Если денег внесено больше, то печатает «Возьмите сдачу» и указывает сумму сдачи. Если денег недостаточно, то печатает об этом сообщение и указывает размер недостающей суммы. Цена книги и имеющаяся в наличии сумма вводятся в рублях: неотрицательные числа, не превышающие 1 000 000 с использованием дробной части для задания копеек.

2.12. Вводится три целых числа, не превышающих по абсолютному значению 30 000. Для этих чисел:

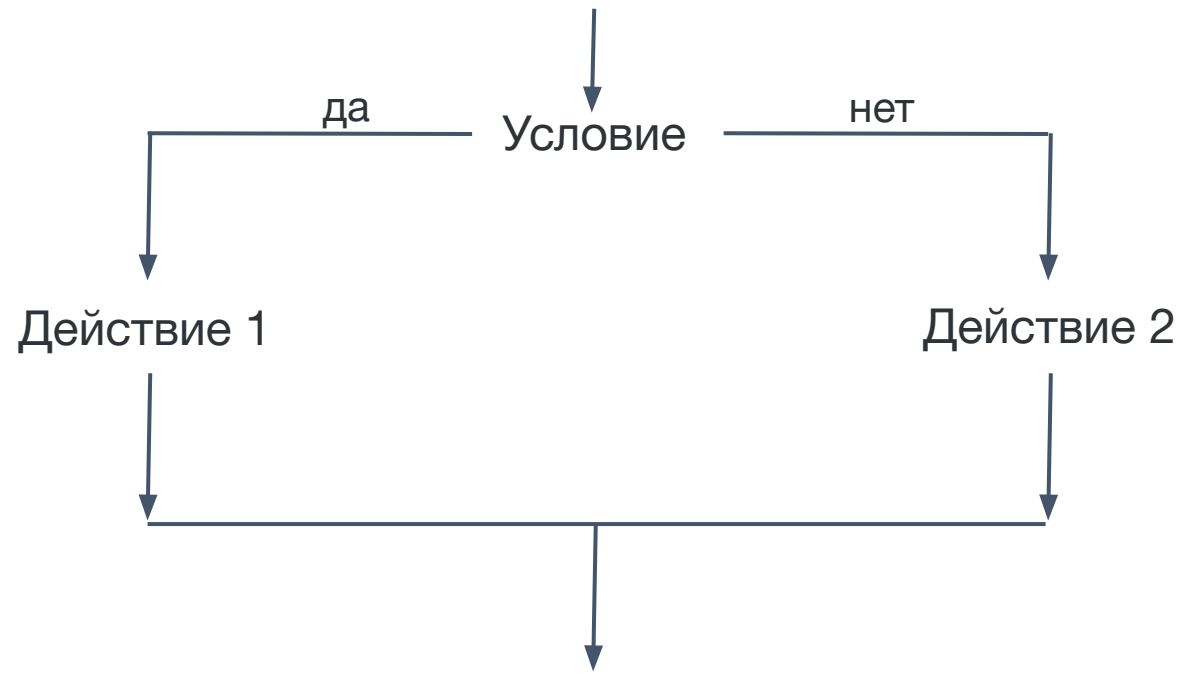
10

# Пример урока

Пройдено понятие ветвления

Пройден оператор ветвления в Паскале

Не изучались логические операторы AND, OR, NOT





# Задача

Вводится два целых числа, **не превышающих по абсолютному значению 30 000**. Для этих чисел определить и сообщить, являются ли оба числа с одним знаком (считать ноль одним знаком с положительным числом).

Необходимо для выбора типа данных. Не нужно проверять корректность введенных чисел, сосредотачиваемся на решении задачи.

# Задача

Вводится два целых числа, не превышающих по абсолютному значению 30 000. Для этих чисел определить и сообщить, являются ли оба числа с одним знаком (считать ноль одним знаком с положительным числом).

## Формируем набор тестовых данных

| На входе: | Ответ:      |
|-----------|-------------|
| 3 6       | один знак   |
| -3 6      | разный знак |
| 3 -6      | разный знак |
| -3 -6     | один знак   |
| 0 6       | один знак   |
| -3 0      | разный знак |

# Задача

Вводится два целых числа, не превышающих по абсолютному значению 30 000. Для этих чисел определить и сообщить, являются ли оба числа с одним знаком (считать ноль одним знаком с положительным числом).

1) Ввод данных: вводится 2 целых числа

Как назвать переменные? **X и Y? A и B?**



# Задача

Вводится два целых числа, не превышающих по абсолютному значению 30 000. Для этих чисел определить и сообщить, являются ли оба числа с одним знаком (считать ноль одним знаком с положительным числом).

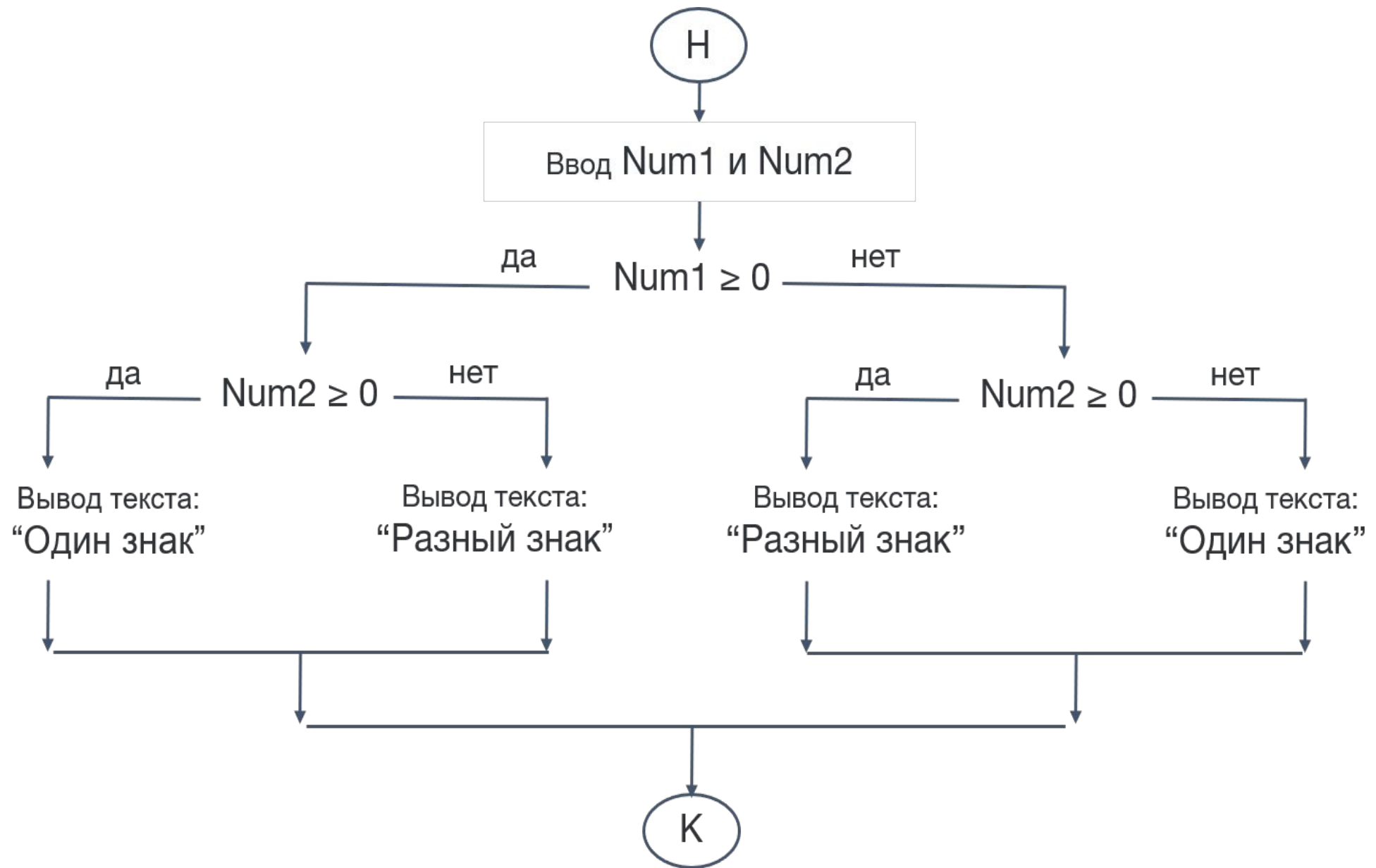
1) Ввод данных: вводится 2 целых числа **Num1** и **Num2**

Имена переменных должны соответствовать  
смыслу!

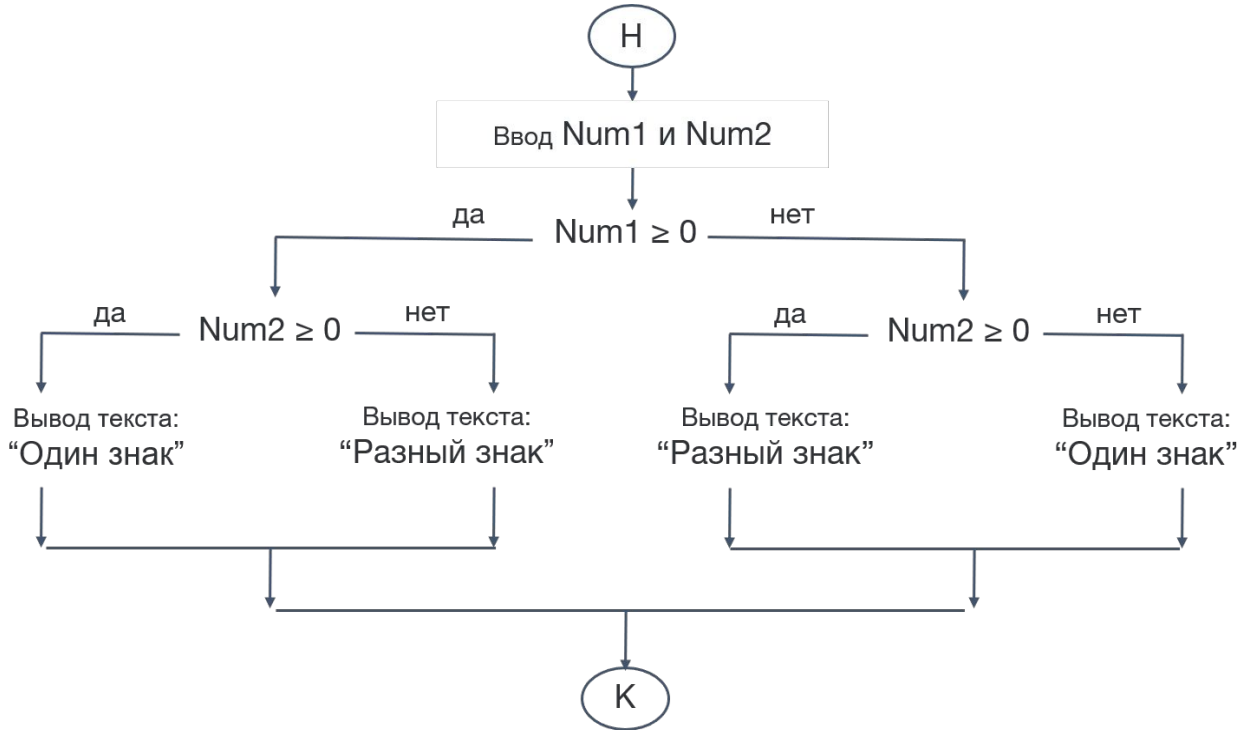
# Задача

Вводится два целых числа, не превышающих по абсолютному значению 30 000. Для этих чисел определить и сообщить, являются ли оба числа с одним знаком (считать ноль одним знаком с положительным числом).

- 1) Ввод данных: вводится 2 целых числа **Num1** и **Num2**
- 2) Формируем решение в визуально-текстовой форме, не привязываясь к языку программирования



# Выполняем тестирование

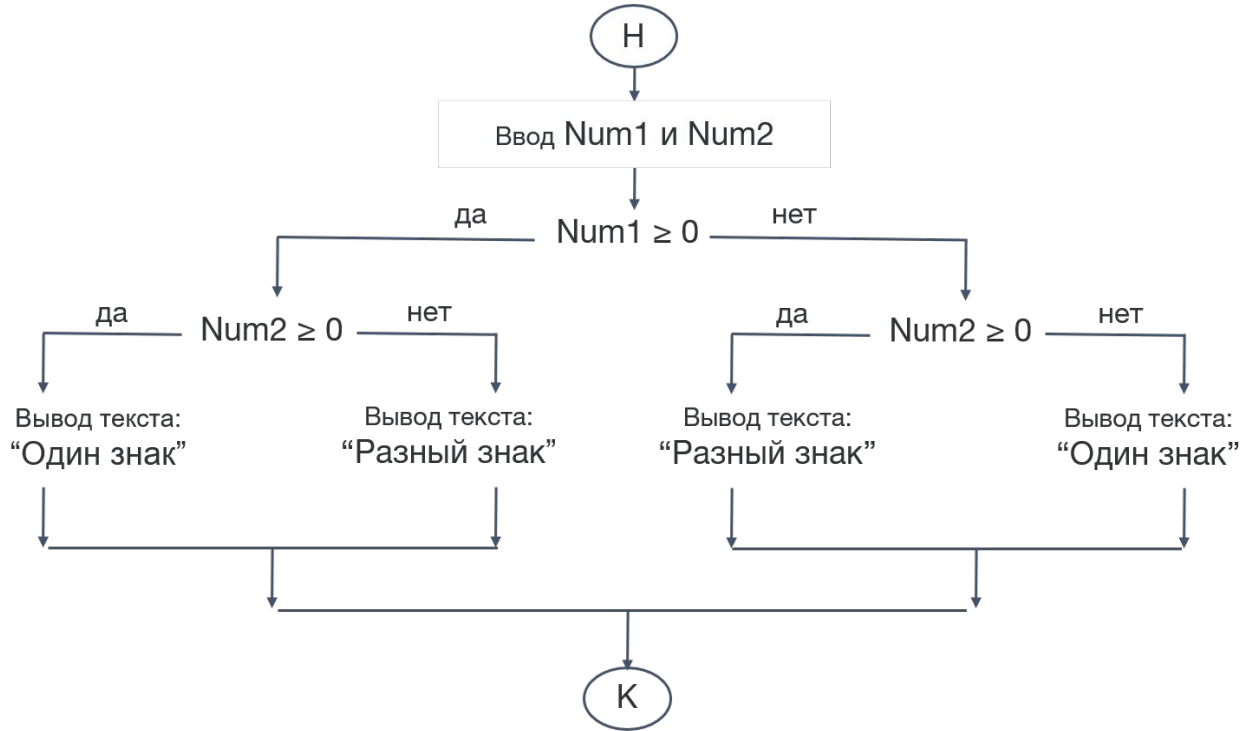


## На входе:

- 1) 3 6
- 2) -3 6
- 3) 3 -6
- 4) -3 -6
- 5) 0 6
- 6) -3 0

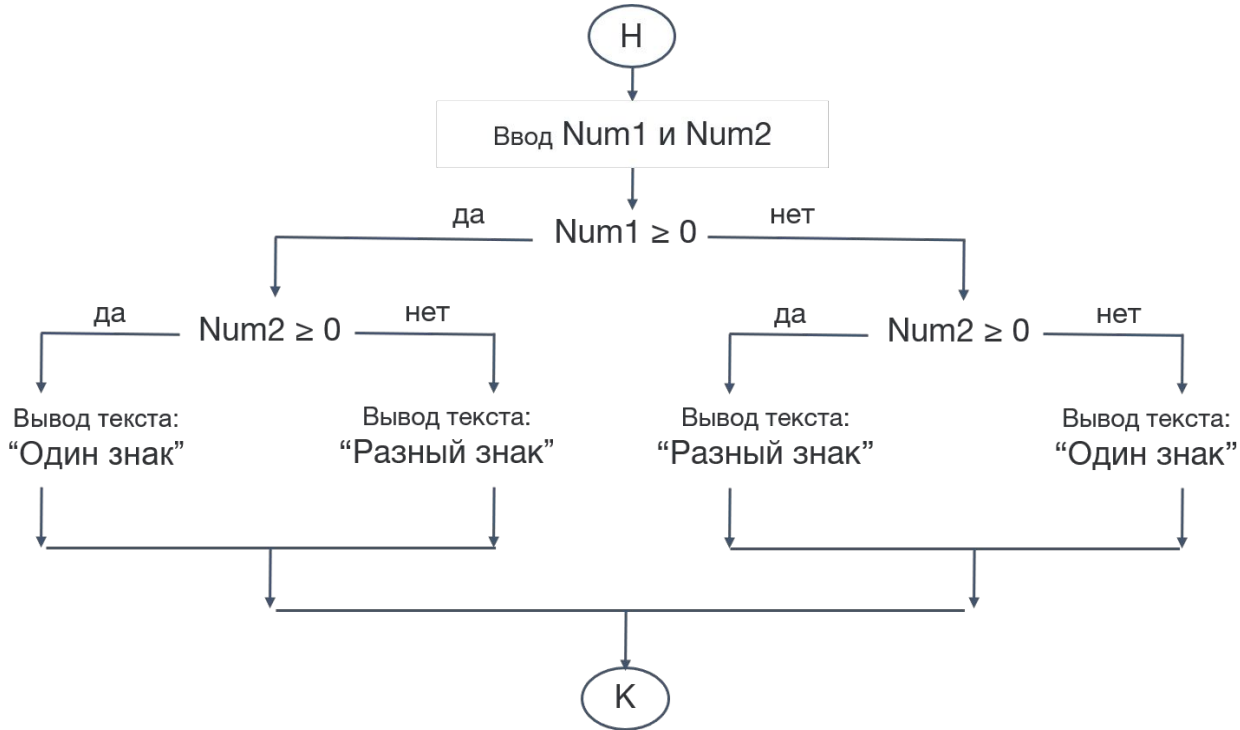
## Ответ:

- один знак  
разный знак  
разный знак  
один знак  
один знак  
разный знак

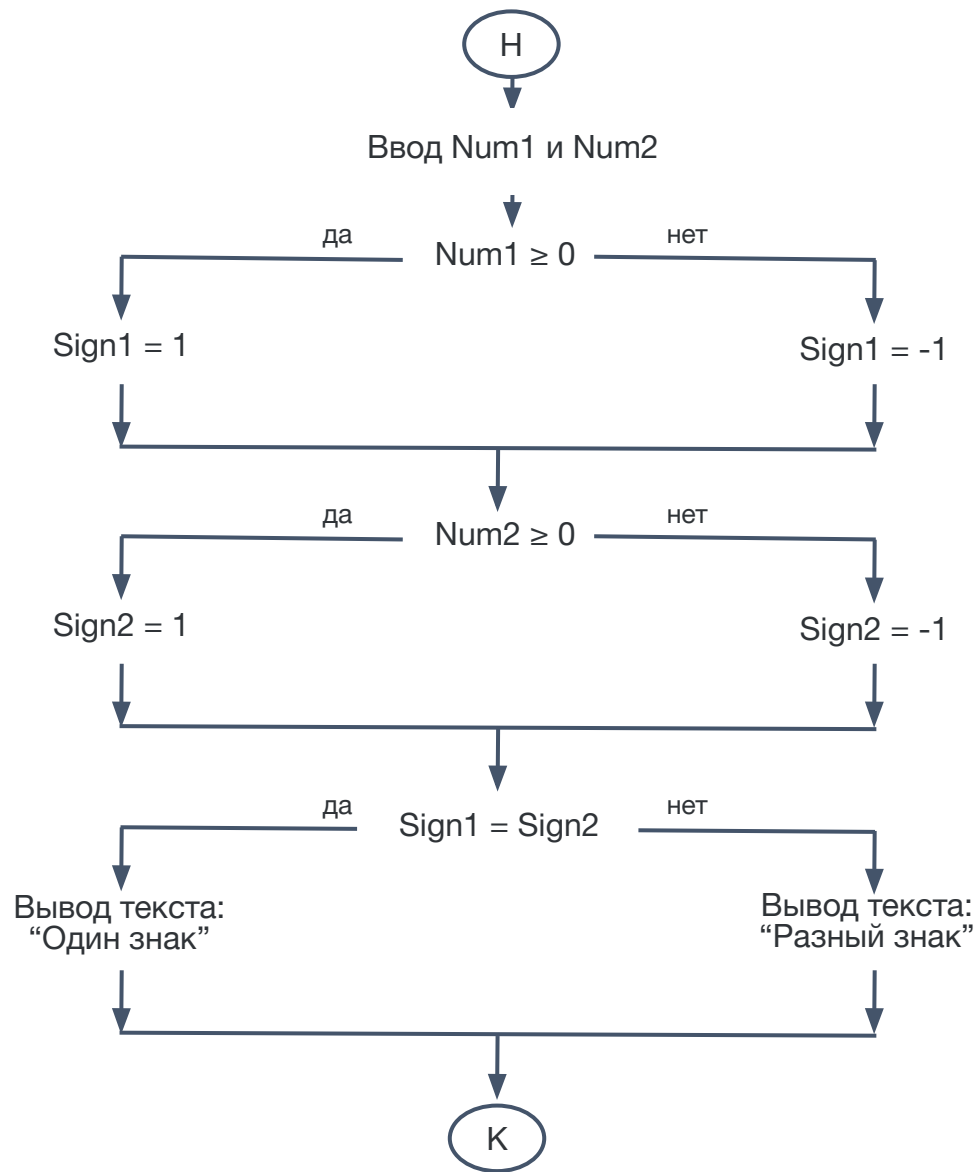


+ Работает верно

Обычно в школе на этом останавливаются



- + **Работает верно**
- **Сложная структура ветвлений ⇒ тяжело читать код**
- **Вывод ответа совмещен с обработкой ⇒ сложность масштабирования трудности совместной разработки**



Вводим переменные “знак”= Sign

Два числа – два знака **Sign1** и **Sign2**.

### Знак 1 числа:

**Sign1 = 1**, если число **Num1** положительное ( $\geq 0$ )

**Sign1 = -1**, если число **Num1** отрицательное

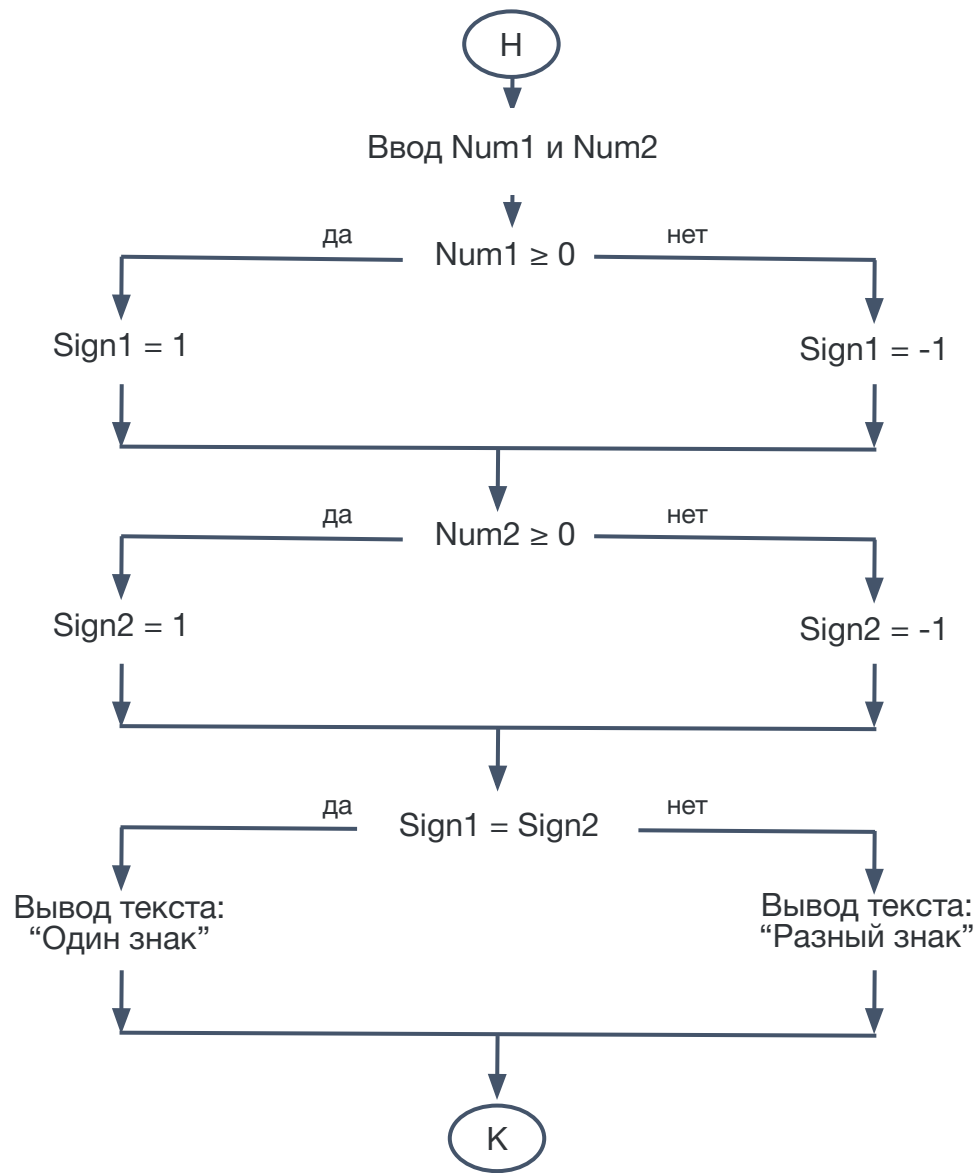
### Знак 2 числа:

**Sign2 = 1**, если число **Num2** положительное ( $\geq 0$ )

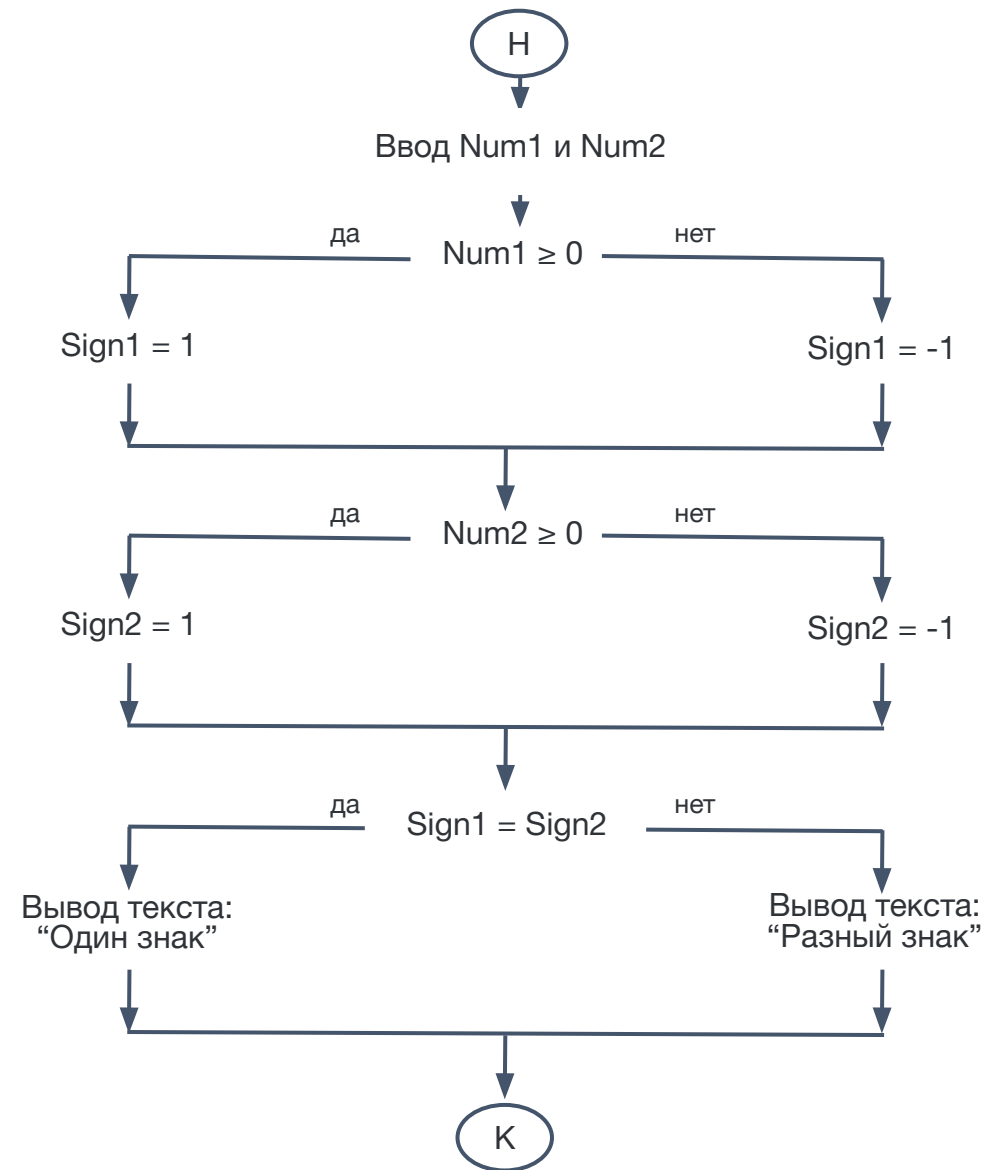
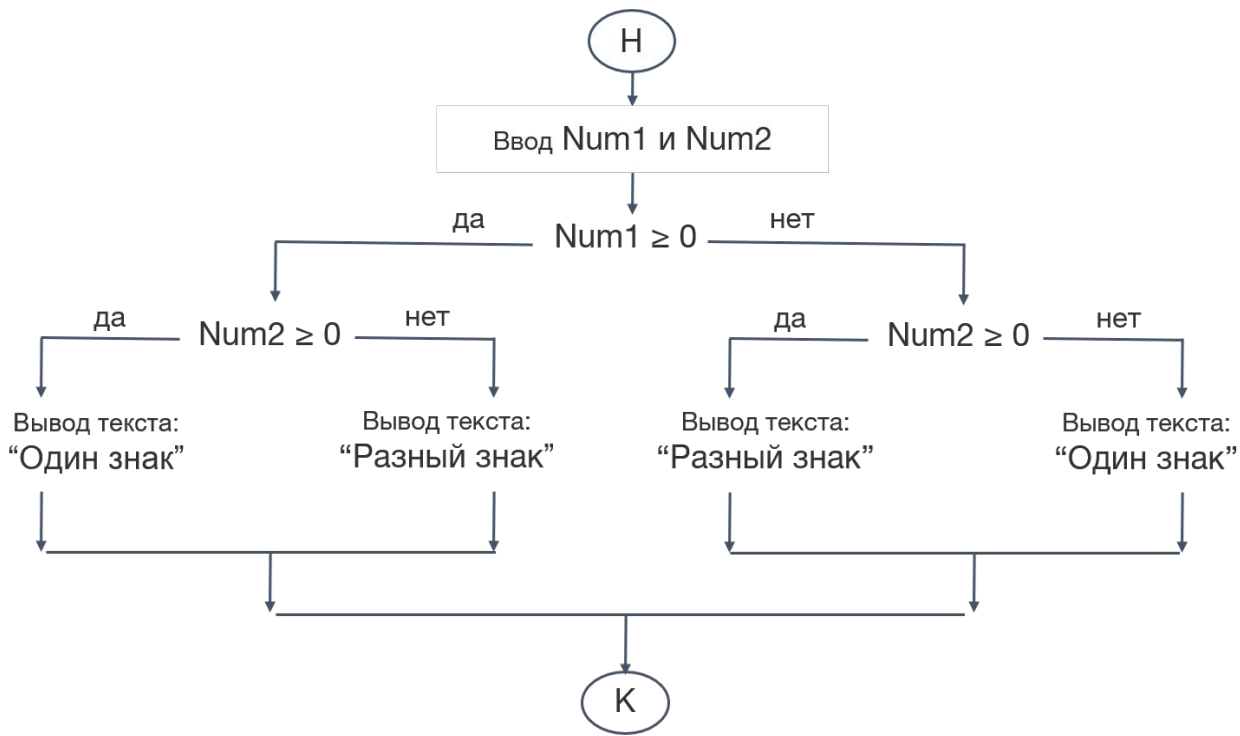
**Sign2 = -1**, если число **Num2** отрицательное

Сравниваем полученные знаки, если у чисел один знак, то и значения знаков совпадают.





- + Работает верно
  - + Проще читать код
  - + Обработка каждого числа однотипна
  - + Проще совместная разработка
  - + Проще масштабирование
- Визуально работа программы не изменилась!!!



Два кандидата на трудоустройство  
представили решение тестового задания.  
**Кто более конкурентоспособен?**

# О выборе языка программирования

| Преимущества использования языка для развития программиста                                                                   | Python | C++ | Free pascal |
|------------------------------------------------------------------------------------------------------------------------------|--------|-----|-------------|
| Формируется понятие переменной как ячейки памяти ограниченных размеров, типов данных (формирование понятия приведения типов) | —      | +   | +           |
| Логические выражения записываются без приведения типов (закрепление тем алгебры логики)                                      | —      | —   | +           |
| Отсутствие автоматического обнуления переменных (формирование привычки инициализации переменных в нужном месте алгоритма)    | —      | +   | +           |
| Отсутствие готовых решений (изучение и оценка эффективности алгоритмов)                                                      | —      | ±   | +           |
| Возможность решения разнообразных прикладных задач                                                                           | +      | ±   | —           |
| Простота установки и настройки среды программирования, использования стандартных операторов                                  | +      | —   | +           |

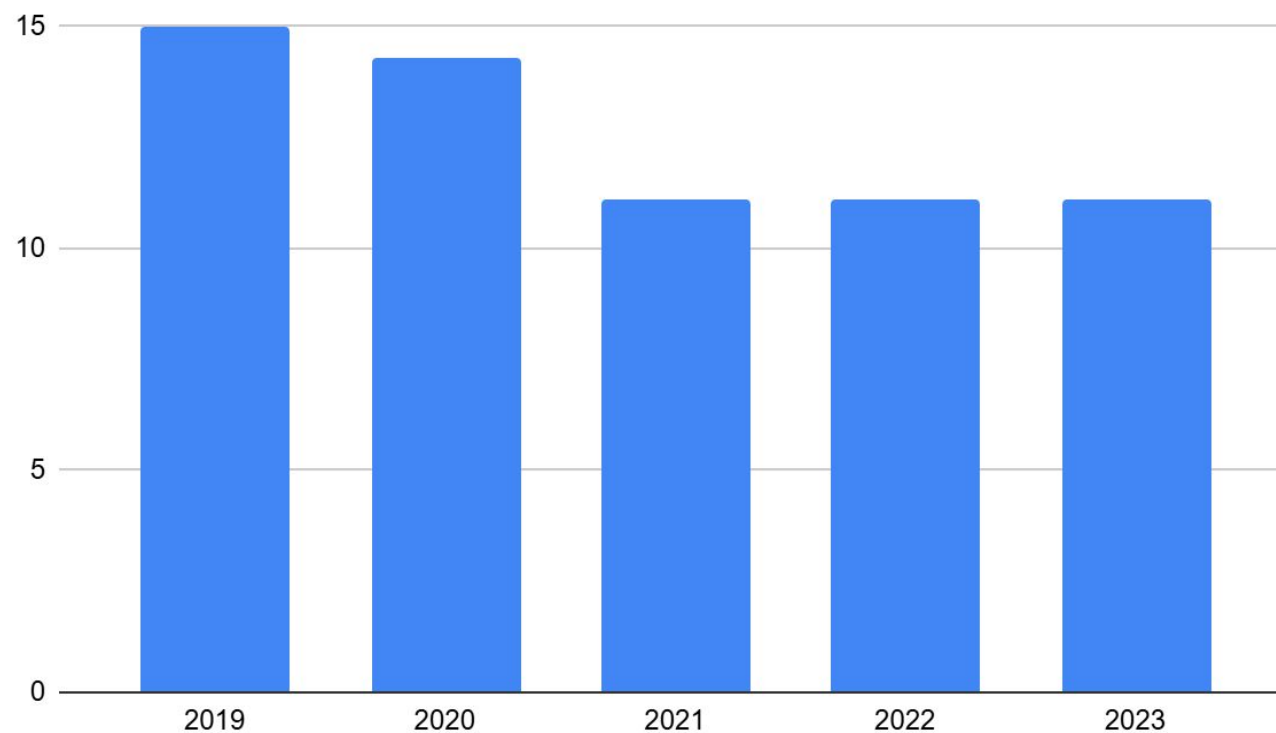
# Предметные результаты

| Описание результата                                                                                                                                              | базовый | продвинутый |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|-------------|
| Соответствие требованиям ФГОС основного общего и среднего общего образования к результатам освоения учебного предмета “Информатика”                              | +       | +           |
| Разработка кода с учётом принятых в команде требований к оформлению / форматированию                                                                             | +       | +           |
| Выделение этапов разработки и структурирование кода с учётом выполняемых обобщенных действий: ввод, валидация, поиск решения, вывод и тестирование каждого этапа | +       | +           |
| Разработка и структурирование кода с использованием подпрограмм и модулей, с учётом уровней абстракции и требований безопасности кода                            | —       | +           |
| Проектирование решения (продукта) с точки зрения эффективности командной разработки и масштабируемости                                                           | —       | +           |

# Результаты применения продуктового подхода

12 из 95 выпускников смогли начать работать после окончания школы.

Работает сразу после окончания 11 класса, %



# Отзывы выпускников

«Было полезно: декомпозиция задач, системный подход... Получить задачу и не просто её сделать, а иметь несколько вариантов решения. Все это (и даже Паскаль) дало сильную базу для самостоятельного развития. Конкурс в Казани помог понять, что программирование – не какая-то оторванная от реальности сфера, а с базовыми навыками можно уже что-то полезное делать.»



**Илья Шулепов**, выпускник 2020 г.,

победитель секции «Программное обеспечение»  
на IV Всероссийском форуме высоких  
технологий kazanforum.doc в г. Казани, 2017 г.,

backend разработчик компании Sibedge.



# Отзывы выпускников

«Перед поступлением в лицей, у меня уже был опыт и работы, я «набирал код» на клавиатуре и считал это работой программиста. Но, проучившись 2 года, я понял, что есть разница между тем, чем я занимался до, и решением инженерных задач в продукте: в лицее нас учили писать код так, чтобы он работал бесперебойно, корректно обрабатывал самые уникальные сценарии и был понятен десяткам программистов после тебя. Именно этот уровень ответственности выделяет продуктового инженера среди других программистов. Освоить новую технологию не составляет труда, самое главное – нам привили навык думать как инженер.»

**Рустам Гайнутдинов**, выпускник 2019 г.,

инженер-программист отдела разработки систем обучения компании iSpring,  
преподаватель Института iSpring, руководитель практик студентов.





# Отзывы выпускников

«Я начал замечать разницу между мной и друзьями из других школ где-то в 9-10 классах. Когда мы больше углублялись в фундаментальные знания, мои друзья использовали программирование, как неизвестный инструмент для достижения непонятно чего. То есть было сильно заметно, что мы в Инфотехе понимали намного больше, чем ребята из других школ.

Мне была важна на начальных этапах связь между тем, что мы делаем, и для чего мы это делаем. Сложные моменты, знания объяснялись или давались на самостоятельное изучение. Но самое главное – мне было понятно, для чего это нужно, и как я могу это потом использовать.

Я начал работать в iSpring, как только пошел на 1 курс в Волгатех. Считаю, что основным фактором для этого являлась практика после 10 класса, когда мне и ещё двум товарищам предоставили возможность сделать продакшен-продукт – личный кабинет для Инфосферы. Я попал в настоящую среду программистов, которые были на несколько голов меня круче, поэтому жадно впитывал все знания, приёмы, поведения, чтобы стать таким же крутым.»

**Андрей Дудин**, выпускник 2019 г., инженер-программист компании iSpring.





# Пишите, будем рады взаимодействию!

Жеребцова Ольга [Olga.Zherebtsova@ispring.com](mailto:Olga.Zherebtsova@ispring.com)

Гарипова Юлия [garipovaye@mail.ru](mailto:garipovaye@mail.ru)

Институт iSpring <http://ispring.institute/>

Лицей «Инфотех» <https://www.infotech12.ru/>

КШ «Инфосфера» <https://isphera.ru/>