



## НОВЫЕ УПРАВЛЕНЧЕСКИЕ ТЕХНОЛОГИИ В ПЕДАГОГИКЕ

### Методика обучения программированию учащихся: проблемы и решения

#### Анна Сергеевна Зуфарова

Старший преподаватель кафедры «Информатика»  
Тихоокеанский государственный университет  
Хабаровск, Россия  
006694@pnu.edu.ru  
 0000-0000-0000-0000

#### Роман Александрович Суходуб

Группа КБ(с)-01  
Тихоокеанский государственный университет  
Хабаровск, Россия  
2020103338@pnu.edu.ru  
 0000-0000-0000-0000

Поступила в редакцию 14.04.2022

Принята 12.05.2022

Опубликована 15.06.2022

 10.25726/j1495-0128-8440-f

#### Аннотация

Актуальность данной темы обусловлена тем, что с развитием компьютерных технологий, в нашей стране не хватает качественных специалистов в области IT технологиях. Особо остро почувствовали в нынешней политической ситуации. Образовательный процесс пытается активно внедрять информационные технологии, но не успевает в ногу со временем. Человечество и передовых странах цифровые технологии выходят на первое место. Сейчас мы себе не можем представить жизнь без смартфона, персонального компьютера или любого программного обеспечения обслуживающее любую отрасль. Оно используется в медицине, в обороне страны, в защите информации, в образовании, в создании кино и мультимедиа, в проектировании космических моделей и во многом другом. Но чтоб его создать нужны специалисты в этой области. Их нужно сначала взрастить, выучить и выпустить в жизнь. Этот процесс долгий и трудоемкий. В данной статье анализируется ситуация с информационными технологиями в образовании в средних и высших учебных заведениях. Предлагаются пути решения этих проблем.

#### Ключевые слова

информатика, программирование, алгоритмизация, информационная культура, образовательный процесс, языки программирования, Python, Pascal.

#### Введение

Актуальность данной темы обусловлена переменами в современной социальной, экономической и политической жизни общества. Цифровизация охватила все сферы деятельности человека от образования до промышленности. В учебных планах содержат дисциплины связанные с информационными технологиями: информатика, ИКТ, цифровая культура, алгоритмизация, программирование, базы данных, информационная безопасность и другие (Как разработчики, 2019).

Проблемы обучению программированию рассматриваются в работах А.П.Ершова, А.Г. Кушниренко, В.М.Монахова, так же в методику обучения программированию внесли вклад М.М.Бежанова, Э.З.Любимский.

Информационные технологии это очень подвижный пласт науки, а особенно в 21 веке. Поэтому рабочие программы должны быть актуальные всегда.

На рынке труда в области информационных технологий требуются не только высокие требования к уровню знаний выпускников вузов, но и опыт труда в этой области. Без владения современных информационных технологий и продвинутого уровня в программировании молодым специалистом трудоустроится тяжело в современном мире. В IT компаниях требуются сразу квалифицированный и опытный специалист, а молодежь без опыта неохотно берут (В России). Так же многие наши талантливые ребята уезжают за границу за опытом и высокооплачиваемой зарплатой. В нынешней ситуации потребность в высококвалифицированных IT-специалистов высока в нашей стране. Не хватает специалистов в области IT- технологий: программирования, обеспечение защиты информации, создание различного программного обеспечения и другое. Это связано с тем, что в 21 веке использование информационных технологий в образовательном процессе сильно отстают от мировых тенденций. Среднее и высшие образовательные учреждения не могут предложить высококвалифицированных IT-специалистов и с колоссальным опытом работы (Кузенкова, 2021).

### **Материалы и методы исследования**

В нашей стране ситуацию с изучением программирования в школах, она выглядит плачевно. Это справедливо не только для России, но и многих развитых стран. В нашем образовании приоритетные точные науки (алгебра, геометрия, физика и другие), гуманитарные науки (история, обществознание и другие). Понятно, здесь проявляются многолетние традиции в учебных заведениях, учебный план, количество часов и другие факты.

Но важный факт в обучении является сам педагог. У многих преподавателей информационных технологий, информатики, основ программирования низкий уровень знаний в этой области, т.е. педагог не компетентен в этой области знаний. И с этим надо бороться как-то. Брать молодые кадры, повышать зарплаты, вводить стимулирующие надбавки. Ежегодно проводить повышения квалификаций педагогов. Так же открывать в вузах по всей Росси, не только в крупных городах Москва или Санкт-Петербург, кафедры от крупных наших IT компаний, например Яндекс, Mail.ru.

Вторая проблема это: информационная база учебных заведений желает лучшего. Старое программное обеспечение, старое компьютерное оборудование и нехватка его на лабораторных работах это тоже является большой проблемой. Учащиеся сидят по 2- 3 ученика за компьютером, что не дает эффекта в познание предмета. Бесперебойное подключение к глобальной сеть интернет через организацию, чтобы использовать виртуальные классы, электронные ресурсы, использовать интернет ресурсы (Бузыкова, 2020).

Преподаватель должен объяснять на пальцах компьютерные технологии, программирование в 21 веке. Даже не во всех кабинетах есть инструменты для пока презентаций. Но как можно на доске показывать программное обеспечение в современном мире? Хочется внедрить видео урок, а у педагога нет возможности показать нюансы и заинтересовать учащихся. Ведь мы знаем человек воспринимает информацию почти 90 % зрением, т.е. визуалью (Зуфарова, 2020).

### **Результаты и обсуждение**

Изучение создание блок-схем и алгоритмов является лакмусовой бумажкой в проявления интереса к данной тематике у учащихся, что значительно повлияет на выбор будущей профессии. Будет базой для дальнейшего изучения узких предметов в IT технологиях.

Был проведен опрос студентов во всех направлениях по поводу знакомства с информационными технологиями на разных этапах обучения – школа и университет.

Результаты опроса студентов показал, что:

1. Обучающие испытывают трудность в области: алгоритмизации, языков программирования: 58 % ответили нет, 42 % - да.

2. Изучали ли вообще языки программирование в школе – 64,5 % -да, 35,5% -нет

3. Нужна ли математика, математическая логика для программиста -80,6% да, 19,4% -нет.

По результатам исследования видно, что существуют трудности у студентов с предметами связанных с программированием, алгоритмизацией. Так же почти у 64,5% опрошенных студентов не изучали основ программирование и алгоритмизации в школе. И это большая проблема для учащихся, которые хотят связать свою жизнь с программированием и созданием программного обеспечения.

Так же 19,4 % учащихся, считают, что знание математики не нужно для программирования. Но математика, логика, алгоритмизация, программирование стоят в одном ряду и связаны между собой цепочкой. Без знания основ математики, логики, алгоритмизации трудно будет понять и написать программный код.

Так же исследование показало, что предмет информатика у 22,6 % опрошенных начинался в 7 классе и программирование преподавалось на разном уровне и на разных этапах обучения в школе. В начальной школе программирование не преподавалось.

Информационную культуру нужно развивать с самого начального обучения, начать с садика, начальной школы и средней школы.

Должно быть ранее обучение школьников алгоритмизации и программированию. Обучение должно быть в игровой форме в садике, например, роботехника. Нужно разработать рабочие программы в этой области, чтоб ребенку было интересно и понятно.

Так же в образовательных учреждениях надо разработать курсы по программированию для начальных классов, проводить дополнительные занятия для учащихся по алгоритмизации, программированию. На начальной стадии в игровой форме, но постепенно усложнять обучение. А в старших классах предмет информатика - алгоритмизация и программирование проходить основательно. Привлекать преподавателей с вузов, IT компаний.

В итоге студентам поступившим на специальности с IT уклоном намного легче и интереснее будет проходить обучающийся процесс. Так же на разных курсах обучения ввести сотрудничество с различными IT компаниями и ввести это в ежедневное расписание, не как трудовую практику, а специально выделить день. Студенты получают нужный опыт, что на выпуске мы получим квалифицированного специалиста и не нужно будет еще ждать год или два, чтобы получить специалиста в данной области.

Студентам первых и иногда вторых курсов трудно переключиться со школьного программирования на профессиональные языки Java, Python, C++, машинные языки программирования. В качестве языка программирования в школах используют Basic, Pascal, алгоритмический язык. Есть даже интересная цитата Эдсгер Вибе Дейкстра (голландский учёный): «Студентов, ранее изучавших Бейсик, практически невозможно обучить хорошему программированию. Как потенциальные программисты они подверглись необратимой умственной деградации» (КАРТАСЛОВ).

Но благодаря ОГЭ, ЕГЭ в школах постепенно вводят изучения профессиональных языков программирования: Python, C ++. На рисунке 1 приведены задания из экзаменов ОГЭ, ЕГЭ. Если рассмотреть эволюцию экзаменов ( ОГЭ, ЕГЭ), то видно как эволюционируют задания. С 2015 года задания на Python входят в ЕГЭ (8 заданий): часть 1 – 5 заданий, часть 2 – 3 задания, а с 2021 благодаря переменам экзамен полностью вынесен в компьютерной форме. Ребята раньше информатику решали на бумажке, а сейчас максимум используют свои навыки в информационных технологиях. В ЕГЭ почти во всех заданиях нужно умение программировать, и в качестве языка программирования используют Python.

Бейсик	Python
<pre>DIM k, s AS INTEGER INPUT s INPUT k IF s &lt;= 2 * k THEN PRINT "ДА" ELSE PRINT "НЕТ" END IF</pre>	<pre>s = int(input()) k = int(input()) if s &lt;= 2 * k: print("ДА") else: print("НЕТ")</pre>
Паскаль	Алгоритмический язык
<pre>var s, k: integer; begin readln(s); readln(k); if s &lt;= 2 * k then writeln ('ДА') else writeln ('НЕТ') end.</pre>	<pre>алг нач цел s, k ввод s ввод k если s &lt;= 2 * k то вывод "ДА" иначе вывод "НЕТ" все кон</pre>
C++	
<pre>#include &lt;iostream&gt; using namespace std; int main() { int s, k; cin &gt;&gt; s; cin &gt;&gt; k; if (s &lt;= 2 * k) cout &lt;&lt; "ДА"; else cout &lt;&lt; "НЕТ"; }</pre>	

Ниже на пяти языках программирования записан рекурсивный алгоритм F.

Бейсик	Python
<pre>SUB F(n) PRINT n IF n &gt; 0 THEN F(n - 1) F(n - 3) END IF END SUB</pre>	<pre>def F(n): print(n) if n &gt; 0: F(n - 1) F(n - 3)</pre>
Паскаль	Алгоритмический язык
<pre>procedure F(n: integer); begin writeln(n); if n &gt; 0 then begin F(n - 1); F(n - 3) end end</pre>	<pre>алг F(цел n) нач вывод n, нс если n &gt; 0 то F(n - 1) F(n - 3) все кон</pre>
Си	
<pre>void F(int n) { cout &lt;&lt; n; if (n &gt; 0) { F(n - 1);</pre>	

Рисунок 1. Задание 6 ОГЭ и 16 ЕГЭ

Проведем сравнительный анализ языка программирования Pascal и Python. Основные алгоритмы :

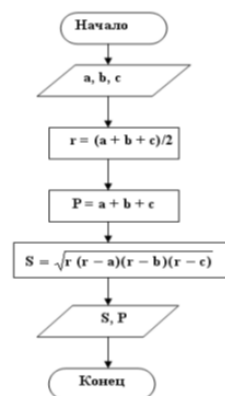
1. Алгоритм линейной структуры,
2. Алгоритм разветвляющейся структуры,
3. Алгоритм циклической структуры.

На рисунке 2 и 3, представлены алгоритм линейной структуры: блок-схема и код программ.

## ПРИМЕР ЛИНЕЙНОГО АЛГОРИТМА НА ЯЗЫКЕ PASCAL

### Вычисление площади и периметра треугольника

```
var S, P, r, a, b, c: real;
begin
write('Введите стороны треугольника ');
readln(a, b, c);
r := (a + b + c)/2;
P := a + b + c;
S := sqrt(r*(r-a)*(r-b)*(r-c));
writeln('Периметр: ', P, 'Площадь: ', S);
end.
```



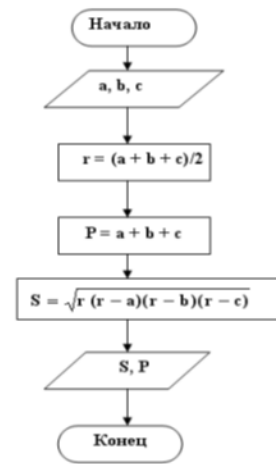
Блок-схема алгоритма

Рисунок 2. Пример линейного алгоритма на языке Pascal

## ПРИМЕР ЛИНЕЙНОГО АЛГОРИТМА НА ЯЗЫКЕ PYTHON

### Вычисление площади и периметра треугольника

```
a = float(input('Введите a: '))  
b = float(input('Введите b: '))  
c = float(input('Введите c: '))  
r = (a + b + c)/2  
P = a + b + c  
S = (r*(r-a)*(r-b)*(r-c)) ** 0.5  
print('Периметр:', P, 'Площадь:', S)
```



Блок-схема алгоритма

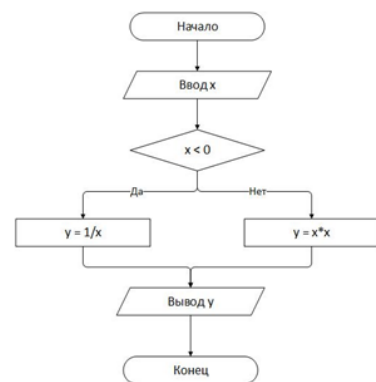
Рисунок 3. Пример линейного алгоритма на языке Python

На рисунке 4 и 5, представлены алгоритм разветвляющейся структуры: блок-схема и код программ.

## ПРИМЕР ВЕТВЛЕНИЯ НА ЯЗЫКЕ PASCAL

Вычисление значения функции  $y = \begin{cases} \frac{1}{x}, & x < 0 \\ x^2, & x \geq 0 \end{cases}$

```
var x, y: real;  
begin  
  write('Введите x: ');  
  readln(x);  
  if (x < 0) then  
    y := 1/x  
  else  
    y := x*x;  
  writeln('y = ', y);  
end.
```



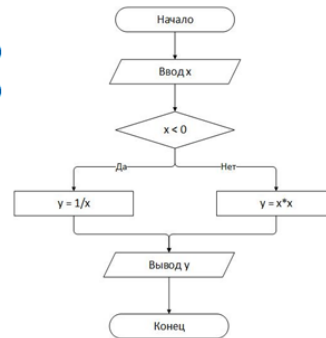
Блок-схема алгоритма

Рисунок 4. Пример разветвляющегося алгоритма на языке Pascal.

## ПРИМЕР ВЕТВЛЕНИЯ НА ЯЗЫКЕ PYTHON

Вычисление значения функции  $y = \begin{cases} \frac{1}{x}, & x < 0 \\ x^2, & x \geq 0 \end{cases}$

```
x = float(input('Введите x: '))
if x < 0:
    y = 1/x
else:
    y = x*x
print('y = ', y)
```



Блок-схема алгоритма

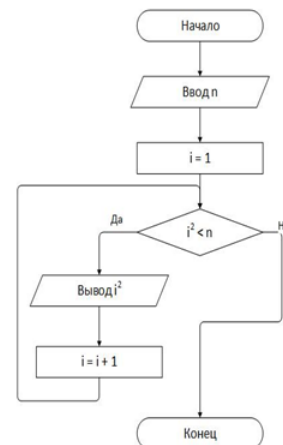
Рисунок 5. Пример разветвляющегося алгоритма на языке Python

На рисунке 6 и 7, представлены алгоритм циклической структуры: блок-схема и код программ.

## ПРИМЕР ЦИКЛА НА ЯЗЫКЕ PASCAL

Вывод квадратов чисел, меньших заданного числа

```
var n, i: integer;
begin
write('Введите n: ');
readln(n);
i := 1;
while (i*i < n) do begin
    writeln (i*i);
    inc(i);
end;
end.
```



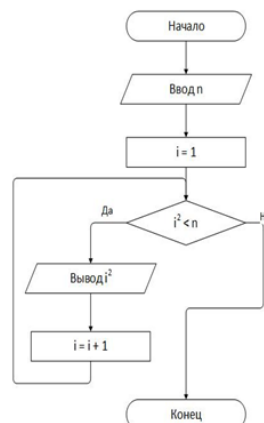
Блок-схема алгоритма

Рисунок 6. Пример циклического алгоритма на языке Pascal.

## ПРИМЕР ЦИКЛА НА ЯЗЫКЕ PYTHON

**Вывод квадратов чисел, меньших заданного числа**

```
n = int(input('Введите n: '))  
i = 1  
while (i*i < n):  
    print(i*i)  
    i += 1
```



Блок-схема алгоритма

Рисунок 7. Пример циклического алгоритма на языке Python

Сделав сравнительный анализ двух языков программирования Pascal и Python, видно, что Python - самый стильный язык программирования. Он не допускает двойного написания кода, т.е. код можно написать только одним способом. В нем отсутствуют лишние конструкции, и это прекрасно видно и более читаемого кода нет ни в одном другом языке программирования.

Вывод о Python: открытый код; простота и лаконичность синтаксиса; наличие мощных библиотек; многоплатформенность; бесплатность; прост в изучении; сообщество поддержки. Это все пункты хорошо подходят, как для начинающих программистов (школьников), продолжающих программистов (студентов) и профессионалов (выпускники вузов, IT специалисты).

Так же в школах сделать ориентир:

1. начальная школа – изучение основ блок-схем и алгоритмизации, как инструмент познания и изучения мира. Очень хорошо подходит линейный и разветвляющийся алгоритмы.
2. Средняя школа – изучение базовых алгоритмов, основы теоретической информатики (система счисления, объем информации, логические операции и другое)
3. Старшая школа – изучение структуры программ, использование процедуры, функции, библиотек. Создание значимых проектов по программированию и другое.

### Заключение

Из выше сказанного можно сделать вывод, что решить проблемы в получении качественного специалиста в IT области нужно :

- в школах нужно увеличить часы изучения предмета информатики, как для профильных классов, так и нет;
- в садиках и начальной школе можно ввести начальные предметы по изучению информатики в игровой форме;
- вузам страны перейти на сотрудничество с IT компаниями для обучения студентов, школьников;
- ежегодно обновлять и добавлять актуальную информацию в рабочие программы;
- развивать законодательство в области образования связанных с IT технологий
- внедрять в процесс профессиональные и современное программное обеспечение для обучения студентов IT направлений;
- проводить для педагогов информационные съезды в области IT технологий, повышения квалификаций.

### Список литературы

1. Бузыкова Ю.С., Зуфарова А.С. Роль информационно-коммуникационных технологий в профессиональном образовании // Современное педагогическое образование. 2020. № 9. С. 84-87.
2. В России возрастает дефицит IT-специалистов, и бизнес берёт инициативу в свои руки. <https://propostuplenie.ru/article/v-rossii-vozrastaet-deficit-it-specialistov-i-biznes-beret-iniciativu-v-svoi-ruki/>
3. Зуфарова А.С. Роль технологии визуализации в учебной информации // Современное педагогическое образование. 2020. № 9. С. 39-41.
4. Как разработчики учат своих детей программировать. <http://itsuhorukov.ru/2019/06/02/kak-razrabotchiki-uchat-svoih-detey-programmirovat-i-nado-li-eto-delat/>
5. КАРТАСЛОВ.РУ. Карта слов и выражений русского языка. <https://kartaslov.ru>
6. Кузенкова Г.В., Шагбазян Д.В., Штанюк А.А., Обучение программированию в средней и высшей школе: проблемы и пути решения // Современные проблемы науки и образования. 2021. № 1. <https://science-education.ru/ru/article/view?id=30514>

### Methods of teaching students programming: problems and solutions


#### Anna S. Zufarova

Senior lecturer of the Department of "Informatics"

Pacific State University

Khabarovsk, Russia

006694@pnu.edu.ru

 0000-0000-0000-0000


#### Roman A. Sukhodub

KB Group(s)-01

Pacific State University

Khabarovsk, Russia


2020100392@pnu.edu.ru

 0000-0000-0000-0000

Received 14.04.2022

Accepted 12.05.2022

Published 15.06.2022

 10.25726/j1495-0128-8440-f

### Abstract

The relevance of this topic is due to the fact that with the development of computer technology, our country lacks high-quality specialists in the field of IT technologies. We felt it especially sharply in the current political situation. The educational process is trying to actively introduce information technology, but does not keep up with the times. For mankind and advanced countries, digital technologies come out on top. Now we cannot imagine life without a smartphone, personal computer or any software serving any industry. It is used in medicine, in the defense of the country, in the protection of information, in education, in the creation of films and cartoons, in the design of space models and much more. But to create it, you need experts in this field. They must first be nurtured, learned and released into life. This process is long and laborious. This article analyzes the situation with information technologies in education in secondary and higher educational institutions. Ways of solving these problems are proposed.



**Маша maKeywords**

computer science, programming, algorithmization, information culture, educational process, programming languages, Python, Pascal.

**References**

1. Buzykova Ju.S., Zufarova A.S. Rol' informacionno-kommunikacionnyh tehnologij v professional'nom obrazovanii // *Sovremennoe pedagogicheskoe obrazovanie*. 2020. № 9. S. 84-87.
2. V Rossii vozrastaet deficit IT-specialistov, i biznes berjot iniciativu v svoi ruki. <https://propostuplenie.ru/article/v-rossii-vozrastaet-deficit-it-specialistov-i-biznes-berjot-iniciativu-v-svoi-ruki/>
3. Zufarova A.S. Rol' tehnologij vizualizacii v uchebnoj informacii // *Sovremennoe pedagogicheskoe obrazovanie*. 2020. № 9. S. 39-41.
4. Kak razrabotchiki uchat svoih detej programmirovat'. <http://itsuhorukov.ru/2019/06/02/kak-razrabotchiki-uchat-svoih-detey-programmirovat-i-nado-li-eto-delat/>
5. KARTASLOV.RU. Karta slov i vyrazhenij russkogo jazyka. <https://kartaslov.ru>
6. Kuzenkova G.V., Shagbazjan D.V., Shtanjuk A.A., Obuchenie programmirovaniju v srednej i vysshej shkole: problemy i puti reshenija // *Sovremennye problemy nauki i obrazovanija*. 2021. № 1. <https://science-education.ru/ru/article/view?id=30514>