

Методические указания.

Циклы

СЮЖЕТНАЯ ЛИНИЯ:

Интернет-магазин «Долголетие» остался доволен выполненным заказом и обратился в ProTeam с ещё одной задачей. Директор «Долголетия» хочет дополнить сайт вводом промокода при оформлении заказа. Пользователь должен иметь возможность вводить промокод до тех пор, пока программа не распознает его как правильный. Разработчики берутся за заказ и осознают необходимость изучения и использования циклической конструкции.

РЕЗЮМЕ:

Цель урока — сформировать у учеников понимание цикла как конструкции, выполняющей команды, пока остаётся истинным логическое выражение (условие).

Занятие начинается с повторения пройденного материала. Затем ученики изучают циклическую конструкцию и оператор `while`. Дети узнают синтаксис и научатся программировать цикл и цикл со счётчиком. Полученные знания применяются при решении сюжетных задач на платформе.

ССЫЛКИ И РЕКВИЗИТ:

- [презентация](#) к уроку;
- задания на платформе ([часть 1](#), [часть 2](#)).

ОБРАЗОВАТЕЛЬНЫЕ РЕЗУЛЬТАТЫ УРОКА

После урока ученик:	Результат достигнут, когда ученики:
• своими словами описывает назначение циклической конструкции; • знает оператор <code>while</code> и оформление цикла; • задаёт условие работы цикла с помощью логического выражения; • выбирает подходящую алгоритмическую конструкцию: условную или циклическую; • своими словами описывает необходимость использования счётчика для подсчёта количества повторов цикла; • уверенно программирует цикл <code>while</code> со счётчиком;	• участвовали в обсуждении и задавали уточняющие вопросы; • осознали необходимость ввода новой конструкции для программирования повторяющегося действия; • справились с сюжетными задачами на платформе; • при возникновении ошибки ученики пытались справиться сами или обращались за помощью к соседям; • на этапе закрепления назвали оператор <code>while</code>, рассказали, как задать условие его работы, для чего может использоваться счётчик и как его запрограммировать.

РЕКОМЕНДУЕМАЯ СТРУКТУРА УРОКА

Время	Этап	Задачи этапа
5 мин 	Сюжетная линия. Обсуждение: «Ввод промокода»	❑ Поставить сюжетную задачу: запрограммировать ввод промокода для магазина «Долголетие» (до тех пор, пока он не будет распознаваться как правильный). ❑ Сформулировать проблему: необходимо запрограммировать повторяющееся выполнение команд при истинности некоторого условия. ❑ Прийти к необходимости изучить циклическую конструкцию и оператор while.
10 мин 	Квалификация	❑ Вспомнить пройденный материал: ❑ сложные логические выражения, операции и их приоритет; ❑ условный оператор и его виды.
15 мин 	«Мозговой штурм»: «Цикл while»	❑ Используя блок-схемы, рассказать о циклической конструкции и сравнить её с условной. ❑ Продемонстрировать синтаксис оператора while (задание набора команд и условия их выполнения). ❑ Разобрать примеры реализации циклических алгоритмов (блок-схемы и соответствующие им программы).
15 мин 	Платформа: «Долголетие: оформление заказа»	❑ Организовать выполнение задания «Долголетие: оформление заказа».
5 мин. 	Перерыв	❑ Помочь ученикам восстановить концентрацию.
15 мин 	«Мозговой штурм»: «Цикл со счётчиком»	❑ Озвучить требование заказчика: подсчитать количество повторов цикла. ❑ Рассказать о счётчике как о переменной, в которой можно накапливать повторы цикла. ❑ Привести примеры задач с использованием цикла со счётчиком.
15 мин 	Платформа: «Долголетие: подсчёт действий»	❑ Организовать выполнение сюжетного задания: «Долголетие: подсчёт действий».
10 мин 	Завершение урока. Рефлексия	❑ Пройти техническое собеседование по темам: цикл while, цикл со счётчиком. ❑ Предложить дополнительное задание и «документацию» на платформе.

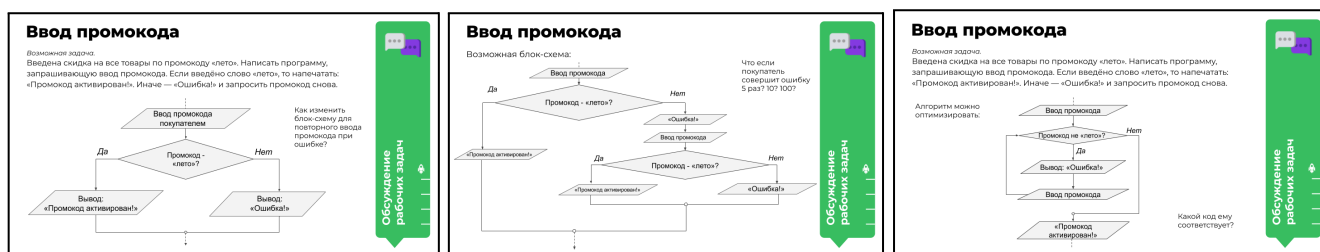
Сюжетная линия. Обсуждение: «Ввод промокода»

(10 мин.)

Откройте презентацию. Разработчикам компьютеры пока не требуются.

«Здравствуйте, коллеги! Вчера Алёна сообщила мне, что директор “Долголетия” высоко оценил наше техническое решение. Теперь он хочет дополнить сайт магазина тематическими акциями. Для этого на сайт нужно добавить возможность вводить промокоды. Обратите внимание, что ввод не должен ограничиваться одной попыткой».

Рассмотрите задачу с вводом промокода «лето». Если пользователь вводит данный промокод, то программа печатает: «Промокод активирован». В остальных случаях печатается слово «Ошибка!» и ввод запрашивается снова.



Обсудите задачу по частям:

- Попросите описать работу программы при вводе одного промокода. Как определить его правильность? Как напечатать подходящее сообщение?
- По условию, если промокод введён неверно, нужно запросить повторный ввод. Можно ли это реализовать имеющимися средствами? Как?
- Что, если пользователь снова введёт промокод с ошибкой? Что, если неверный промокод будет введён 10 раз? 100? Как доработать алгоритм?
- Опишите инструмент (алгоритмическую конструкцию), необходимый для оптимизации программы.

Обобщите предложения разработчиков:

«Чтобы составить алгоритм, запрашивающий ввод до тех пор, пока не будет получен верный промокод, и запрограммировать его, нужно ввести новую алгоритмическую конструкцию — цикл».

Сформулируйте цель рабочего дня и анонсируйте его содержание.

Квалификация

(10 мин.)

Организируйте подтверждение квалификации. В текущий рабочий день она проходит по материалам прошлого и позапрошлого «мозговых штурмов».

«Мозговой штурм»: «Цикл while»

(15 мин.)

С помощью презентации введите новую алгоритмическую конструкцию — цикл. Раскройте определение цикла и покажите, что он используется для выполнения набора действий до тех пор, пока остаётся истинным некоторое логическое выражение (условие).

Как запрограммировать повтор действий?

Мы знаем, как запрограммировать условие — предложение, которое может быть истинным или ложным. Теперь освоим конструкцию, повторяющую команды до тех пор, пока условие остаётся истинным.

```

        graph TD
            Start([Ввод промодка]) --> Cond{Промодка не «Китая»?}
            Cond -- Да --> Out([Вывод: «Счастливая»])
            Out --> End([Промодка активирована])
            Cond -- Нет --> Start
            
```

«Мозговой штурм»

Цикл

— это команда, выполняющая указанные действия до тех пор, пока остаётся истинным логическое выражение (условие).

Пример:
Цикл выполняет действие А до тех пор, пока истинно логическое выражение

```

        graph TD
            Start([Логическое выражение истинно?]) -- Да --> Act[Выполнить действие А]
            Act --> Start
            Start -- Нет --> End([Конец])
            
```

«Мозговой штурм»

Обсудите задачу с выигрышным номером карты. Она состоит из двух частей: составления алгоритма и написания кода. При переходе от первой части ко второй продемонстрируйте оформление цикла с помощью служебного слова while.

- Предложите разработчикам составить алгоритм (блок-схему). Он аналогичен алгоритму из обсуждения начала рабочего дня.
- Продемонстрируйте скриншот ожидаемой работы программы и предложите составить программу с циклом while.
- Изучите код программы. Как будет работать эта программа, если последовательно вводить: «14155», «498722», «45626»? Сколько раз сработает цикл?

Обсудите задачу на составление программы для сбора отзывов покупателей:

- Спросите, какая алгоритмическая конструкция потребуется для решения задачи. Почему необходимо использовать именно цикл, а не условный оператор?
- Попросите назвать логическое выражение, которое будет помещено в цикл и перечислить команды, которые будут выполняться в теле цикла.
- Изучите код программы. Можно ли его оптимизировать? Как сработает программа, если последовательно ввести «Вкусно!», «Не понравилось обслуживание», «Посоветую друзьям», «off»? Сколько раз сработает цикл?

Обсудите задачу со скидкой 10% на все товары.

Рассмотрим задачу

Задача 3 Составить программу, выводящую стоимость всех покупок со скидкой 10%. Программа запрашивает ввод покупок до тех пор, пока не введён «0». Затем выводится стоимость покупок со скидкой 10%.

```

        Введите цену (0 - завершить):
        >>> 100
        Введите цену (0 - завершить):
        >>> 150
        Введите цену (0 - завершить):
        >>> 80
        Введите цену (0 - завершить):
        >>> 0
        Стоимость без скидки: 330
        Стоимость со скидкой: 297.0
        
```

«Мозговой штурм»

Рассмотрим задачу

Задача 3 Составить программу, выводящую стоимость всех покупок со скидкой 10%. Программа запрашивает ввод покупок до тех пор, пока не введён «0». Затем выводится общая стоимость всех покупок со скидкой 10%.

```

        price = int(input("Введите цену (0 - завершить):"))
        total_price = 0
        while price != 0:
            total_price += price
            price = int(input("Введите цену (0 - завершить):"))
        print("Стоимость без скидки:", total_price)
        total_price = total_price * 0.9
        print("Стоимость со скидкой:", total_price)
        
```

«Мозговой штурм»

Перед тем, как продолжить :

- Что напечатает программа, если сразу ввести «0»?
- Какую цену без скидки напечатает программа, если последовательно ввести: «100», «120», «215», «0»?
- Какую цену со скидкой напечатает программа, если последовательно ввести: «50», «100», «50», «0»?
- Что произойдёт, если пользователь напечатает «250 руб»?

«Мозговой штурм»

Примечание. Скорее всего, ваши ученики не проходили проценты в школе. Поэтому сразу на примере поясните, что для того, чтобы подсчитать цену со скидкой 10%, её нужно умножить на 0,9.

- Спросите, какая алгоритмическая конструкция потребуется для решения задачи. Почему необходимо использовать именно цикл? Попросите назвать логическое выражение, которое будет помещено в цикл.
- Чтобы сделать скидку 10% на все покупки, нужно подсчитать общую стоимость и умножить её на 0,9. Как посчитать сумму всех покупок? На каком этапе нужно применить скидку?
- Изучите код программы. Затем задайте вопросы на понимание текста программы.

Подведите итоги и переходите к выполнению сюжетного задания на платформе.

Платформа: «Долголетие: оформление заказа»

(15 мин.)

Организуйте выполнение заданий по программированию циклов на платформе. Напомните разработчикам, что во все уровни встроена автопроверка.

Ответы на задания вы найдёте в конце методических указаний.

Перерыв

(5 мин.)

Отвлеките разработчиков от компьютеров. Цель перерыва — переключить внимание и размяться. Организуйте одну из [предложенных физических активностей](#).

«Мозговой штурм»: «Цикл со счётчиком»

(15 мин.)

Начните обсуждение с пожелания заказчика. Программа, считывающая промокод работает исправно, но нужно взять на контроль количество попыток ввода. Если пользователь ввёл промокод, потратив аномально большое количество попыток, то, возможно, он использовал мошенническую программу подбора кодов. Такие случаи нужно отслеживать.

На примере подобной задачи введите счётчик как переменную, накапливающую число шагов цикла. Покажите, что счётчик может использоваться не только для подсчёта всех шагов цикла, но и для подсчёта шагов, на которых выполнялось дополнительное условие.



Обсудите задачу с вводом промокода «новый год» и подсчётом попыток.

- От какого условия будет зависеть работа цикла? В какие части программы нужно вставить ввод данных?

- Где должна быть объявлена переменная-счётчик и где будет меняться её значение? Опишите изменения значений счётчика, если пользователь последовательно вводит: «день рождения», «масленица», «новый год».
- Изучите код программы. Как будет меняться значение счётчика, если пользователь введёт несколько раз один и тот же неправильный промокод? («радость», «мандарины», «радость», «новый год»). Что напечатает программа при завершении работы?

Обсудите задачу с акцией «Цены пополам».

Рассмотрим задачу

Задача 2 Составить программу, запрашивающую цены товаров до тех пор, пока не будет введён «0» и вычисляющей общую сумму покупок. Если число товаров чётное, то включается акция «Цены пополам», и вся сумма делится пополам. В завершение программа печатает сумму и сдачу.

```

Введите цену (0 - остановит):
>>> 100
Введите цену (0 - остановит):
>>> 200
Введите цену (0 - остановит):
>>> 50
Введите цену (0 - остановит):
>>> 100
Введите цену (0 - остановит):
>>> 0
Акция «Цены пополам»
К оплате: 225.0
                    
```

Как посчитать количество покупок?

«Мозговой штурм»

Рассмотрим задачу

Задача 2 Составить программу, запрашивающую цены товаров до тех пор, пока не будет введён «0» и вычисляющей общую сумму покупок. Если число товаров чётное, то включается акция «Цены пополам», и вся сумма делится пополам. В завершение программа печатает сумму и сдачу.

```

price = int(input('Введите цену (0 - остановит):'))
amount = 0
total_price = 0
while price != 0:
    total_price += price
    amount += 1
    price = int(input('Введите цену (0 - остановит):'))
if amount % 2 == 0:
    print('Акция «Цены пополам»')
    total_price = total_price / 2
print('К оплате:', total_price)
                    
```

«Мозговой штурм»

Перед тем, как продолжить :

1. Что напечатает предыдущая программа, если последовательно ввести: «50», «120», «80», «0»?
2. Сколько раз сработает цикл, если пользователь последовательно введёт: «35», «20», «0»?
3. Можно ли создать бесконечно работающий цикл? Если да, то приведи пример такого цикла.



«Мозговой штурм»

- Как посчитать общее количество товаров, если оно заранее неизвестно? Как посчитать общую стоимость всех товаров, если цены товаров вводятся по очереди? На каком этапе нужно определить, положена ли скидка, и вывести окончательную стоимость товара?
- Изучите код программы. Затем задайте вопросы на понимание текста программы.

Подведите итоги обсуждения и переходите к выполнению задания на платформе.

Платформа: «Долголетие: подсчёт действий»

(15 мин.)

Организуйте выполнение оставшихся заданий для магазина «Долголетие». Во все уровни встроена автопроверка.

Ответы на задания вы найдёте в конце методических указаний.

Завершение рабочего дня. Рефлексия

(10 мин.)

С помощью презентации подведите итоги рабочего дня. Проведите техническое интервью с вопросами по материалам «мозгового штурма».

Предложите дополнительные задания для повышения эффективности и документацию с теорией.

Ответы на задания

Задание 1. «Долголетие: оформление заказа»

1.1. Правильные варианты ответов:

Отметьте верные утверждения о цикле while Правильных ответов может быть несколько.	☒ Цикл while выполняет одну и ту же последовательность действий, пока истинно указанное условие.
	☒ Цикл while может использоваться, когда невозможно определить сколько раз нужно повторить действия.
	☐ Все команды и действия в цикле while пишутся без отступов друг под другом.
	☐ Когда выполнение программы доходит до цикла while, тогда сначала выполняется тело цикла, а потом проверяется условие.
	☒ В качестве условия цикла while может использоваться как простое, так и составное логическое выражение.

1.4. Программа после исправления ошибок:

```
time = int(input('Введите текущее время в часах:'))
while time >= 10 and time < 24:
    print('Мы открыты')
    time = int(input('Введите текущее время в часах:'))
print('Мы закрыты. Часы работы: с 10 до 24.')
```

1.5. Завершенная программа:

```
while input('Введите промокод:') != 'life':
    print('Этот промокод недействителен.')
print('Промокод принят.')
```

1.6. Завершенная программа:

```
promo = input('Введите промокод:')
while promo != 'life' and promo != 'health':
    promo = input('Этот промокод недействителен. Попробуйте снова:')
print('Промокод принят.')
```

1.7. Текст программы сбора отзывов:

```
text = input('Введите ваш отзыв:')
while text != 'off':
    print('Спасибо, ваш отзыв принят!')
    text = input('Введите ваш отзыв:')
```

1.8. Текст программы (автоподсчет суммы покупок):

```
price = int(input('Стоимость товара (0 - покупок больше нет):'))
cost = price
while price != 0:
```

```
price = int(input('Стоимость товара (0 - покупок больше нет):'))
cost += price
print('Стоимость всех покупок: ' + str(cost))
```

Доп задание 1.1.

3	<pre>total = 88 while total % 2 == 0: total /= 2</pre>
5	<pre>sale = 5 while sale < 100: sale *= 2</pre>
10	<pre>i = 0 while i < 10: i += 1</pre>

Доп задание 1.2

```
total = int(input('Введите стоимость покупок:'))
while total != 0:
    print('К оплате: ' + str(total * 0.9))
    total = int(input('Введите стоимость покупок:'))
```

Доп задание 1.3

```
text = input('Введите текст бегущей строки:')
while len(text) > 0:
    print(text[0])
    text = text[1:]
```

Задание 2. «Магазин: рекомендация товаров 2»**2.1.** Верны следующие варианты ответов:

Выбери все программы, в которых тело цикла выполнится 4 раза.

☒	<pre>sale = 1 while sale < 10: sale += 2</pre>
☒	<pre>counter = 4 while counter > 0: counter -= 1</pre>
☐	<pre>i = 0 while i < 10: i += 1</pre>
☐	<pre>total = 50 while total > 20: total -= 2</pre>

2.3. Завершенная программа:

#Первые 3 покупателя получают скидку

```
count = 0
while count < 3:
    card = input('Введите номер карты:')
    print('Поздравляем! Вы получили скидку 10%.')
    count += 1
print('Скидки закончились.')
```

2.4. Программа:

#Подсчёт категорий товаров

```
category = input('Категория (end - завершить):')
count = 0
while category != 'end':
    count += 1
    category = input('Категория (end - завершить):')
print('Всего категорий товаров: ' + str(count))
```

2.5. Программа:

```
attempts = 0
promo = ''
while attempts < 3 and promo != 'fresh':
    promo = input('Введите промокод:')
    attempts += 1
if promo == 'fresh':
    print('Принято с попытки №', attempts)
```

2.6. Программа:

#автомат для выдачи талонов

```
code = input('Введите 0 - получить талон, 1 - выключить аппарат:')
```

```

number = 0
while code != '1':
    if code == '0':
        number += 1
        print('талон номер ' + str(number))
    code = input('Введите 0 - получить талон, 1 - выключить аппарат:')

```

Доп задание. 2.1:

```

total = int(input('Введите сумму:'))
while total % 2 == 0:
    total = total / 2
print('К оплате: ' + str(total))

```

Доп задание 2.2:

```

n = int(input('Введите высоту:'))
i = 0
while i < n:
    print(' ' * (n - i - 1) + '*' * (2 * i + 1))
    i += 1

```

Дополнительное задание. «Магазин: доп задание»**1.1.** Верные варианты ответов:

Выбери все программы, в которых нет ошибок.

☒

```

counter = 0
while counter < 10:
    if counter == 3:
        print('Подарок')
    else:
        print('Скидка')
    counter += 1

```

☒

```

count = 0
while count < 10:
    if count == 3:
        print('Подарок')
    count += 1

```

☐

```

total = 10
while total < 10:
    total += 1
    if total == 3:

```

1.3. Код программы без ошибок:

```

category = input('Категория (stop - завершить):')
while category != 'stop':
    if category == 'мясные изделия':
        print('Скидка 10%.')
    elif category == 'напитки':
        print('Скидка 30%.')
    else:

```

10

```
print('Скидок нет.')  
category = input('Введите категорию продуктов:')
```

1.4. Код программы:

```
category = input('Категория (off - завершить):')  
while category != 'off':  
    cost = int(input('Сумма:'))  
    if category == 'молочные продукты':  
        print('Скидка 10%. К оплате: ' + str(cost * 0.9))  
    elif category == 'выпечка':  
        print('Скидка 30%. К оплате: ' + str(cost * 0.7))  
    else:  
        print('К оплате: ' + str(cost))  
    category = input('Введите категорию продуктов:')  
print('Касса закрыта.')
```