

Программирование на Python

Урок №6

План урока:

1. Визуализация циклов
2. Разбор примера
3. Самостоятельная работа

Визуализация циклов:

Для лучшего понимания процесса работы циклов, необходимо уметь визуализировать их работу. Сегодня мы изучим один из таких методов. Когда мы говорим о том, что цикл выполнится, к примеру, 5 раз, а затем завершит работу, мы сразу понимаем, что имеется ввиду. Но необходимо точно понимать что конкретно будет выполнено за каждый проход цикла. Для этого нам понадобится сразу два инструмента визуализации, которые мы знаем и один новый.. Первый знакомый нам инструмент — это блок-схема алгоритма. Второй — табличка списка. Неизвестный нам инструмент это табличка ввода и вывода. Давайте разберем на практике одну из задач и составим для нее блок-схему и табличку списка.

Разбор примера:

Задача, которую мы разберем. звучит так:
Есть предприятие, на котором работает несколько работников. Если в конце месяца, работник изготовит более 20 изделий, то он получает премию. Если же меньше, то премию он не получает. В начале вводится количество работников, затем их результаты на

конец месяца. Нужно вывести сообщение вида “Работник номер такой-то не получает/получает премию”.

Составим блок-схему и табличку списка для этой задачи, причем пока что только для части ввода результатов и помещения их в список. В дальнейшем, для краткости, мы больше не будем рассматривать часть задания с помещением данных в список так подробно, так как практически везде она будет выглядеть одинаково с точки зрения алгоритма. В случае, если у учеников возникают с этим проблемы, то рекомендуется для каждого задания делать и эту часть.



i	
spisok[i]	

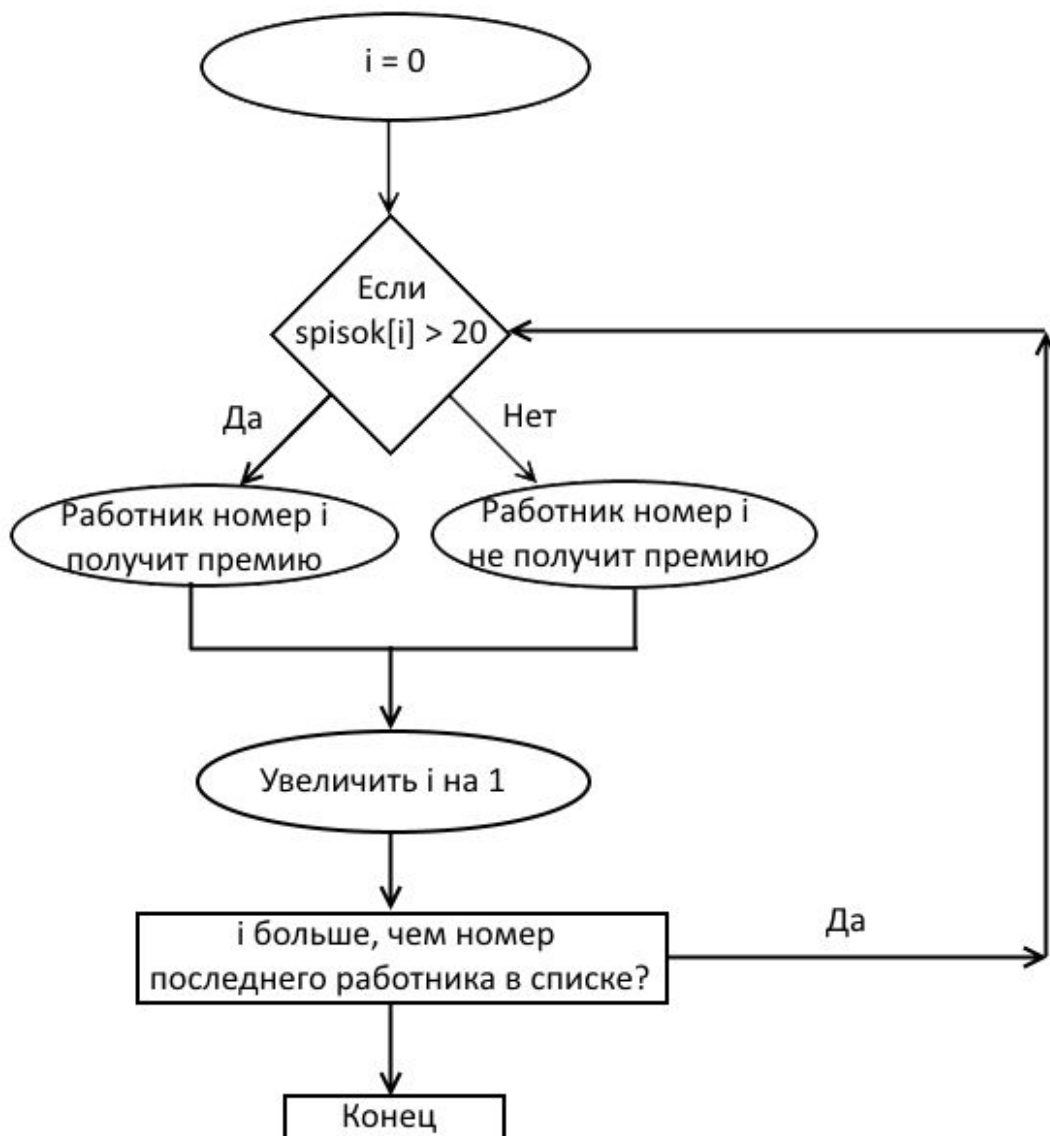
Ввод	Вывод

Пусть, количество работников на предприятии будет 4 человека, а их результаты будут 10, 24, 1, 60. Попробуем с помощью нашего алгоритма и известных данных заполнить наши таблички.

i	0	1	2	3
spisok[i]	10	24	1	60

Ввод	Вывод
4, 10, 24, 1, 60	

Пока что столбец вывода пуст, так как наша программа пока что ничего не выводит на экран. Напишем в виде блок-схемы оставшуюся часть задачи. Блок-схема для этой части будет написана куда более подробно.



Попробуем теперь прогнать наш список через алгоритм и посмотреть, что будет в блоке вывода.

Ввод	Вывод
4, 10, 24, 1, 60	При $i = 0$ "Работник номер 0 не получает премию" При $i = 1$ "Работник номер 1 получает премию" При $i = 2$ "Работник номер 2 не получает премию" При $i = 3$ "Работник номер 3 получает премию"

Теперь составим по имеющимся блок-схемам код на языке Python.

```

print("Введите количество работников")
chis_rabotnikov = int(input())
spisok = []
i = 0
while i < chis_rabotnikov:
    print("Введите результаты работника")
    result = int(input())
    spisok.append(result)
    i = i + 1

```

Вот часть для алгоритма ввода. А теперь добавим к ней код по второй блок-схеме.

```

print("Введите количество работников")
chis_rabotnikov = int(input())
spisok = []
i = 0
while i < chis_rabotnikov:
    print("Введите результаты работника")
    result = int(input())
    spisok.append(result)
    i = i + 1
i = 0
while i < chis_rabotnikov:
    if spisok[i] > 20:
        print("Работник номер", i, "получает премию")
    else:
        print("Работник номер", i, "не получает премию")
    i = i + 1

```

Самостоятельная работа:

1. http://pythontutor.ru/lessons/while/problems/list_of_squares/
2. http://pythontutor.ru/lessons/while/problems/seq_sum/
3. http://pythontutor.ru/lessons/while/problems/seq_avg/
4. http://pythontutor.ru/lessons/lists/problems/increasing_neighbours/
5. http://pythontutor.ru/lessons/lists/problems/same_sign_neighbours/
6. http://pythontutor.ru/lessons/lists/problems/maximal_element/

Тема	Время с начала занятия, мин
Визуализация циклов	10
Разбор примера	30
Самостоятельная работа	90