#### Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра информатики

Дисциплина: Методы трансляции

## ОТЧЁТ по лабораторной работе на тему

Определение модели языка. Выбор инструментальной языковой среды

Выполнил Студент гр. 053502 Шаргородский И.С.

Проверил Ассистент кафедры информатики Гриценко Н.Ю.

### СОДЕРЖАНИЕ

1. Цель работы	3
2. Подмножество языка программирования	
2.1 Числовые и строковые константы	
2.2 Типы переменных	
2.3 Операторы цикла	
2.4 Условные операторы	
3. Инструментальная языковая среда	
Приложение А (информационное) Код программ	
, , , , , , , , , , , , , , , , , , ,	_

#### 1. Цель работы

Необходимо определить подмножество языка программирования (типы констант, переменных, операторов и функций). В подмножество как минимум должны быть включены:

- числовые и текстовые константы;
- 3-4 типа переменных;
- операторы цикла (do...while, for);
- условные операторы (if...else, case).

Определение инструментальной языковой среды, т.е. языка программирования и операционной системы для разработки включает:

- язык программирования с указанием версии, на котором ведётся разработка (напр. Python 3.7);
- операционная система (Windows, Linux и т.д.), в которой выполняется разработка;
  - компьютер (PC / Macintosh).
- В отчете по лабораторной работе дается полное определение подмножества языка программирования, тексты 2-3-х программ, включающих все элементы этого подмножества. Приводится подробное описание инструментальной языковой среды.

#### 2. Подмножество языка программирования

В качестве подмножества языка программирования выбран Python.

Python — высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода. Синтаксис ядра Python минималистичен. В то же время стандартная библиотека включает большой объём полезных функций.

#### 2.1 Числовые и строковые константы

```
-3, -2, -1, 0, 1, 2, 3 (int);
3.5, -2.7 (float литералы);
"", "hello" (str литералы);
2+3j, 0+5j, 2j, -3-5j (complex литералы);
u"", u"hello" (unicode литералы);
```

#### 2.2 Типы переменных

Руthon поддерживает динамическую типизацию, то есть тип переменной определяется только во время исполнения. Поэтому вместо «присваивания значения переменной» лучше говорить о «связывании значения с некоторым именем». В Python имеются встроенные типы: булевый, строка, Unicode-строка, целое число произвольной точности, число с плавающей запятой, комплексное число и некоторые другие. Из коллекций в Python встроены: список, кортеж (неизменяемый список), словарь, множество и другие. Все значения являются объектами, в том числе функции, методы, модули, классы.

#### 2.3 Операторы цикла

– while - выполняет тело цикла до тех пор, пока условие цикла истинно (см. рисунок 1).

```
>>> i = 5
>>> while i < 15:
... print(i)
... i = i + 2
```

Рисунок 1 - Пример цикла while

– for - выполняет тело цикла, совершая определенные действия на каждой итерации, и проверяя условие выхода (см. рисунок 2).

```
>>> for i in 'hello world':
... print(i * 2, end='')
```

Рисунок 2 - Пример цикла for

– continue - начинает следующий проход цикла, не исполняя оставшееся тело цикла (см. рисунок 3).

```
>>> for i in 'hello world':
... if i == 'o':
... continue
... print(i * 2, end='')
```

Рисунок 3 - Пример использования continue

– break - прерывает исполнение цикла (см. рисунок 4).

```
>>> for i in 'hello world':
... if i == 'o':
... break
... print(i * 2, end='')
```

Рисунок 4 - Пример использования break

#### 2.4 Условные операторы

- if - если условие истино, то выполняется блок кода после if (см. рисунок 5).

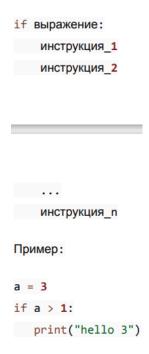


Рисунок 5 - Пример конструкции if

– связка if-else – если условие истино, то выполняется блок кода, который идет после if, иначе выполняется блок кода, который идёт после else (см. рисунок 6).

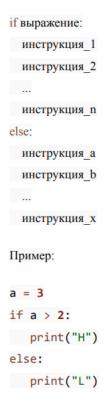


Рисунок 6 - Пример конструкции if-else

#### 3. Инструментальная языковая среда

В качестве языковой среды выбран язык программирования C++. Разработка основана на работе с операционной системой Windows на PC.

компилируемый, статически типизированный язык программирования общего назначения. Поддерживает такие парадигмы программирования, как процедурное программирование, объектноориентированное программирование, обобщённое программирование. Язык имеет богатую стандартную библиотеку, которая включает распространённые контейнеры и алгоритмы, ввод-вывод, регулярные выражения, поддержку многопоточности и другие возможности. С++ сочетает свойства как высокоуровневых, так и низкоуровневых языков. В сравнении с его предшественником — языком С — наибольшее внимание объектно-ориентированного обобщённого уделено поддержке И программирования.

# ПРИЛОЖЕНИЕ А (информационное) Код программ

1. Нахождения факториала числа, введенного пользователем:

```
num = int(input("Enter a number: "))
factorial = 2 + 2 * 2 + 2
if num < 0:
    print("Sorry, factorial does not exist for negative numbers")
elif num == 0:
    print("The factorial of 0 is 1")
else:
    for i in range(1,num):
        factorial = factorial*i
print("The factorial of", num, "is", factorial)</pre>
```

2. N-ое число Фибоначи:

```
\begin{aligned} &\text{num} = \text{int(input("Enter the number of terms: "))} \\ &\text{print("Fibonaci sequence:")} \\ &a = 0 \\ &b = 1 \\ &\text{for i in range(1, num + 1):} \\ &\text{print(" ", a)} \\ &c = a + b \\ &a = b \\ &b = c \end{aligned}
```