

Министерство образования Республики Беларусь  
Учреждение образования  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

КАФЕДРА ИНФОРМАТИКИ

Отчёт по лабораторной работе №1

По теме “Определение модели языка. Выбор инструментальной языковой  
среды.”

Выполнил:  
студент гр. 053504  
Пекутько Д.Л.

Проверил:  
Гриценко Н.Ю.

Минск 2023

## Содержание

<b>1. Цель работы</b>	<b>3</b>
<b>2. Подмножество языка программирования</b>	<b>4</b>
<b>2.1 Типы данных</b>	<b>4</b>
<b>2.2 Основные конструкции языка</b>	<b>4</b>
<b>3. Инструментальная языковая среда</b>	<b>6</b>
<b>Приложения. Код программ</b>	<b>7</b>

## 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

### 2.1 Типы данных

В разрабатываемом языке программирования будет присутствовать сильная типизация. Это означает, что любые действия со значениями, имеющими разные типы данных, требуют явного преобразования типов.

Ниже представлены встроенные типы данных:

- int: -100, 4, 88, 4534 (от MIN\_INT до MAX\_INT в python3).
- float: -332.40, 5.0, 100.00003, (от sys.float\_info.min до sys.float\_info.max в python3)
- str: "hello some", "multiline \n strings"
- bool: true, false
- any: системный тип, который нужен для связки встроенных функций и переменных языка python 3.

### 2.2 Основные конструкции языка

Ниже представлены основные конструкции языка:

- объявления переменных:

```
my_var int;  
my_var2 int = 30;
```

- комплексные выражения

```
my_var2 = some_var + 2 * (3 + (4 - 5)) * (-8) / 100;
```

- логические операторы

```
res bool = 4 and 0 or not 28;
```

- условные конструкции

```
if a > 3 {}  
else if a == 2  
    a = 3;  
else {  
    a = 42;  
}
```

## ● ЦИКЛЫ

```
i int = 0;

loop {
    i = i + 1;
    if a > 3 stop;
    else next;
}

loop i int, 1..10, 2 {
    a = a + 3;
}
```

## ● функции

```
def mult |n1 int, n2 int| -> int {
    ! n1 * n2;
}

mult_res int = [mult 2, 3];
```

### 3. Инструментальная языковая среда

В качестве языковой среды выбран язык программирования Python 3.10.7. Разработка основана на работе с операционной системой Linux Ubuntu 22.10.

Python — высокоуровневый язык программирования, отличающийся эффективностью, простотой и универсальностью использования. Он широко применяется в разработке веб-приложений и прикладного программного обеспечения, а также в машинном обучении и обработке больших данных. За счет простого и интуитивно понятного синтаксиса является одним из распространенных языков для обучения программированию.

Ubuntu — это разрабатываемая сообществом, основанная на ядре Linux операционная система, которая идеально подходит для использования на персональных компьютерах, ноутбуках и серверах. Она содержит все необходимые программы, которые вам нужны: программу просмотра Интернет, офисный пакет для работы с текстами, электронными таблицами и презентациями, программы для общения в Интернет и много других.

## Приложения. Код программ

Ниже представлено несколько программ на исходном языке:

- факториал

```
def fact |n int| -> int {  
    if n <= 1 ! 1;  
    else ! [fact n - 1] * n;  
}  
  
[shown [int_to_str [fact 10]]]; # 3628800
```

- n-ое число фиббоначи

```
def fib |n int| -> int {  
    if n <=2 ! 1;  
    ! [fib n - 2] + [fib n - 1];  
}  
  
[shown [int_to_str [fib 25]]]; # 75025
```

- вывод последовательности чисел фиббоначи:

```
def fib_seq |n int| -> int {  
    f1 int = 1;  
    f2 int = 2;  
    loop i int, 0..n-2-1 {  
        sum int = f1 + f2;  
        f1 = f2;  
        f2 = sum;  
        [shown [int_to_str f2]];  
    }  
}  
  
[fib_seq 25];
```