# Отчёт по Лабораторной работе №3

Отчёт об ошибке:

Заголовок: Ошибка при вычислении Network .

Подробное описание: Ошибка возникает при вычислении Network при вводе /2 маски.

Вывод: Неверный адрес.

Решением является удаление в функции def uint (закомментировал строку для видимости

**#\_\_lshift\_\_ = lambda self, other: uint[length\_in\_bits](super(uint[length\_in\_bits], self).\_\_lshift\_\_(other) & max\_value),)**

**def uint(length\_in\_bits):**

**max\_value = (1<<length\_in\_bits) - 1**

**\_type = type(**

**'uint[{}]'.format(length\_in\_bits),**

**(int,),**

**dict(**

**\_\_slots\_\_ = (),**

**\_\_repr\_\_ = lambda self: "{}('{}')".format(type(self).\_\_name\_\_, self),**

**\_\_invert\_\_ = lambda self: type(self)(self ^ max\_value),**

**# \_\_lshift\_\_ = lambda self, other: uint[length\_in\_bits](super(uint[length\_in\_bits], self).\_\_lshift\_\_(other) & max\_value),**

**)**

**)**

Библиотеки: благодаря обширному функционалу питона, здесь мы обошлись без импорта библиотек.

Также главной особенностью этой лабы состоит в том, что здесь классы сильно упрощают решение задачи

**class DefaultDict(dict):**

**def \_\_init\_\_(self, default\_factory=lambda key: raise\_f(KeyError(key))):**

**self.default\_factory = default\_factory**

**def \_\_missing\_\_(self, key):**

**value = self.default\_factory(key)**

**self[key] = value**

**return value**

Создание главного меню:

**def uint(length\_in\_bits):**

**max\_value = (1<<length\_in\_bits) - 1**

**\_type = type(**

**'uint[{}]'.format(length\_in\_bits),**

**(int,),**

**dict(**

**\_\_slots\_\_ = (),**

**\_\_repr\_\_ = lambda self: "{}('{}')".format(type(self).\_\_name\_\_, self),**

**\_\_invert\_\_ = lambda self: type(self)(self ^ max\_value),**

**# \_\_lshift\_\_ = lambda self, other: uint[length\_in\_bits](super(uint[length\_in\_bits], self).\_\_lshift\_\_(other) & max\_value),**

**)**

**)**