

Zadanie 3.1

3.1.1.

```
x=2; y=3;
if(x>y):
    result=x;
else:
    result=y;
```

Kod jest poprawny składniowo, ale zapisanie go w postaci poniżej byłoby bardziej zgodne ze standardami przyjętymi w Pythonie.

```
x=2
y=3
if(x>y):
    result=x
else:
    result=y
```

3.1.2.

Kod poniżej nie jest poprawny syntaktycznie.

```
for i in "querty": if ord(i)<100: print i
```

Zapis poniżej jest wystarczający do wykluczenia błędów syntaktycznych:

```
for i in "querty":
    if ord(i)<100: print i
```

Zapis poniżej jest zgodny ze standardami w Pythonie:

```
for i in "querty":
    if ord(i)<100:
        print i
```

3.1.3

Kod poniżej jest poprawny syntaktycznie.

```
for i in "axby": print ord(i) if ord(i) < 100 else i
```

Zadanie 3.2

3.2.1

L = L.sort()

Instrukcja ta nie przypisuje posortowanej tablicy do L, ponieważ L.sort() nie zwraca wartości, tylko sortuje tablicę L. Przez co po wykonaniu tej instrukcji zmienna L jest pusta. Poprawne byłoby zastosowanie L.sort() co posortowałoby tablice L.

3.2.2

x, y = 1, 2, 3

Instrukcja przypisania w tym wypadku jest błędna, gdyż po lewej stronie mamy 2 zmienne a po prawej 3 wartości, przez co interpreter nie wie co zrobić z trzecią wartością.

3.2.3

X = 1, 2, 3 ; X[1] = 4

Instrukcja aktualizacji wartości w tuple X jest błędna, gdyż ta struktura jest niezmienna.

3.2.4

X = [1, 2, 3] ; X[3] = 4

Instrukcja aktualizacji wartości w liście X jest błędna, gdyż lista numerowana jest od 0 do n-1 (gdzie n to ilość elementów w liście), przez co X[3]=4 próbuje zaktualizować wartość poza listą, listy są strukturami, których rozmiar można zmieniać ale nie poprzez powyższy przykład, w tym wypadku X.append(4) dodałoby 4 na koniec listy X.

3.2.5

X = "abc" ; X.append("d")

X jest typu String, obiekty tego typu nie posiadają metody append().

3.2.6

map(pow, range(8))

Funkcja map stosuje funkcję podaną jako pierwszy argument do elementów podanych w drugim argumencie. Funkcja pow() przyjmuje 2 argumenty przez co nie jest obsługiwana przez funkcję map().