



# Форматы представления данных в компьютере

---

Для вопросов по курсу:

[natalya.shevskaya@moevm.info](mailto:natalya.shevskaya@moevm.info)

Префикс в теме письма [CS\_23XX]

*Шевская Наталья Владимировна  
СПбГЭТУ “ЛЭТИ”, ФКТИ, МОЭВМ*

# Little-endian и Big-endian

---



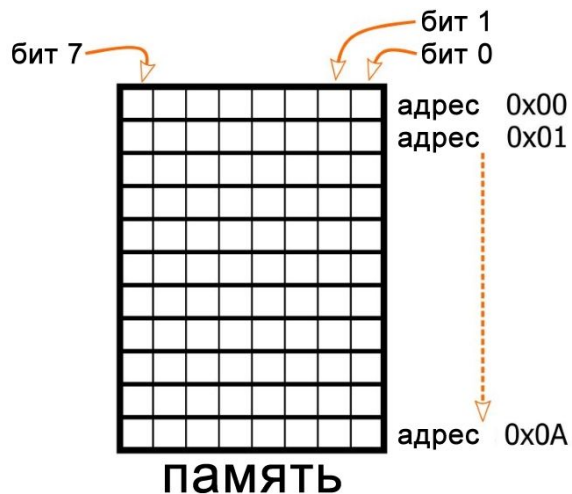
# Little-endian и Big-endian

	Low address				High address			
Address	0	1	2	3	4	5	6	7
Little-endian	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Big-endian	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0
Memory content	0x11	0x22	0x33	0x44	0x55	0x66	0x77	0x88
64 bit value on Little-endian				64 bit value on Big-endian				
0x8877665544332211				0x1122334455667788				

- при работе с многобайтными данными (какой байт считать первым -- младшим, какой последним -- старшим?)



# Прямой и обратный порядок байтов



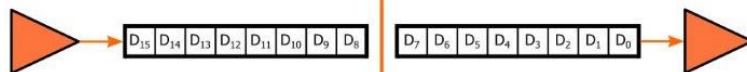
прямой порядок  
big endian

D <sub>31</sub>	D <sub>30</sub>	D <sub>29</sub>	D <sub>28</sub>	D <sub>27</sub>	D <sub>26</sub>	D <sub>25</sub>	D <sub>24</sub>	0x00
D <sub>23</sub>	D <sub>22</sub>	D <sub>21</sub>	D <sub>20</sub>	D <sub>19</sub>	D <sub>18</sub>	D <sub>17</sub>	D <sub>16</sub>	0x01
D <sub>15</sub>	D <sub>14</sub>	D <sub>13</sub>	D <sub>12</sub>	D <sub>11</sub>	D <sub>10</sub>	D <sub>9</sub>	D <sub>8</sub>	0x02
D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>	0x03

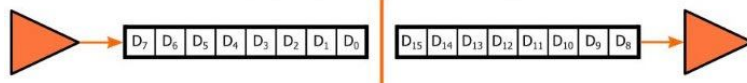
обратный порядок  
little endian

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>	0x00
D <sub>15</sub>	D <sub>14</sub>	D <sub>13</sub>	D <sub>12</sub>	D <sub>11</sub>	D <sub>10</sub>	D <sub>9</sub>	D <sub>8</sub>	0x01
D <sub>23</sub>	D <sub>22</sub>	D <sub>21</sub>	D <sub>20</sub>	D <sub>19</sub>	D <sub>18</sub>	D <sub>17</sub>	D <sub>16</sub>	0x02
D <sub>31</sub>	D <sub>30</sub>	D <sub>29</sub>	D <sub>28</sub>	D <sub>27</sub>	D <sub>26</sub>	D <sub>25</sub>	D <sub>24</sub>	0x03

обратный порядок little endian  
(первым передается младший байт)



прямой порядок big endian  
(первым передается старший байт)



# Сравнение чисел с заданной ТОЧНОСТЬЮ

---



## Сравнение чисел с плавающей точкой

Пример:

Что будет выведено на экран?

```
a = 0.1 + 0.2
```

```
if a == 0.3:
```

```
    print("Числа равны")
```

```
else:
```

```
    print("Числа не равны")
```

# Кодирование символов алфавита

---

# Формат представления символов на компьютере

- ASCII — 7-битовая кодировка, доступно 128 символов.

В Python есть функция `ascii(obj)`, которая возвращает строковое представление объекта с экранированными не-ASCII символами.

```
>>> ascii('привет')
```

- Unicode — 16-битовая кодировка, доступно 65 536 символа.

в Python доступны функции:

```
>>> ord('A')
```

```
>>> chr(95)
```



# Побитовые (поразрядные) операции

---



# Поразрядные операции

Операнд	Описание
	Побитовый OR
^	Побитовый XOR
&	Побитовый AND
<<, >>	Смещения
~x	Побитовый NOT

Чем выше приоритет оператора, тем ниже он находится в таблице и тем раньше он выполняется в смешанных выражениях.



# Источники

- Ч. Петцольд “Код”
- Э. Таненбаум “Архитектура Компьютера”
- <https://stepik.org/course/253> Курс на Stepik “Введение в Архитектуру ЭВМ”

Для вопросов по курсу:  
`natalya.shevskaya@moevm.info`

---

Префикс в теме письма [CS\_23XX]