

Отчёт о работе Multi-threaded MergeSort

Данный алгоритм работает следующим образом :

- 1) Сначала запускается основная функция алгоритма, разделяющая массив на равные части и запускающаяся рекурсивно в новых потоках уже на массивах меньшего размера.
- 2) Когда размер массива доходит до какого-то достаточно малого числа, запускается стандартный однопоточный qsort. Таким образом, ограничивается число потоков для оптимизации времени и эффективности работы программы.
- 3) Далее отсортированные массивы сливаются. Используется функция, разделяющая массив на части и создающая новые потоки, запускающие рекурсивно эту функцию.
- 4) Для малых массивов запускается стандартная функция merge обычного mergeSort.
- 5) Для регулирования максимального числа потоков используется семафор.

In [24]:

```
import matplotlib.pyplot as plt
import numpy as np
%matplotlib inline
```

In [25]:

```
p      =      [2,   4,   8,  16,  24,  32,  64]

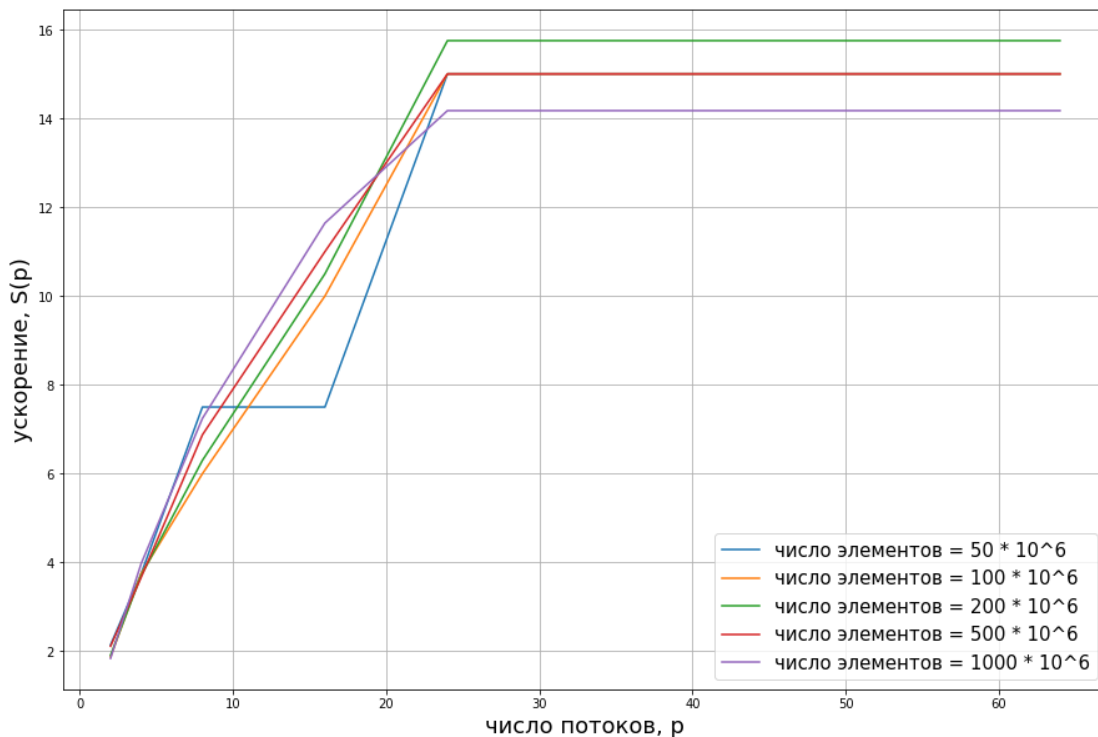
T1 = np.array([7,   4,   2,   2,   1,   1,   1])
T2 = np.array([16,  8,   5,   3,   2,   2,   2])
T3 = np.array([33,  17,  10,   6,   4,   4,   4])
T4 = np.array([78,  45,  24,  15,  11,  11,  11])
T5 = np.array([177, 82,  45,  28,  23,  23,  23])

t1 = 15
t2 = 30
t3 = 63
t4 = 165
t5 = 326
```

Построим графики ускорения $S(p)$ для различных значений количества элементов массива

In [26]:

```
plt.figure(figsize = (15, 10))
plt.plot(p, t1/T1, label=r'число элементов = 50 * 10^6')
plt.plot(p, t2/T2, label=r'число элементов = 100 * 10^6')
plt.plot(p, t3/T3, label=r'число элементов = 200 * 10^6')
plt.plot(p, t4/T4, label=r'число элементов = 500 * 10^6')
plt.plot(p, t5/T5, label=r'число элементов = 1000 * 10^6')
plt.xlabel('число потоков, p', fontsize = 18)
plt.ylabel('ускорение, S(p)', fontsize = 18)
plt.legend(fontsize=15, loc=4)
plt.grid()
plt.show()
```



Построим графики эффективности $E(p)$ для различных значений количества элементов массива

In [28]:

```
plt.figure(figsize = (15, 10))
plt.plot(p, (t1/T1)/p, label=r'число элементов = 50 * 10^6')
plt.plot(p, (t2/T2)/p, label=r'число элементов = 100 * 10^6')
plt.plot(p, (t3/T3)/p, label=r'число элементов = 200 * 10^6')
plt.plot(p, (t4/T4)/p, label=r'число элементов = 500 * 10^6')
plt.plot(p, (t5/T5)/p, label=r'число элементов = 1000 * 10^6')
plt.xlabel('число потоков, p', fontsize = 18)
plt.ylabel('эффективность, E(p)', fontsize = 18)
plt.legend(fontsize=15, loc=1)
plt.grid()
plt.show()
```

