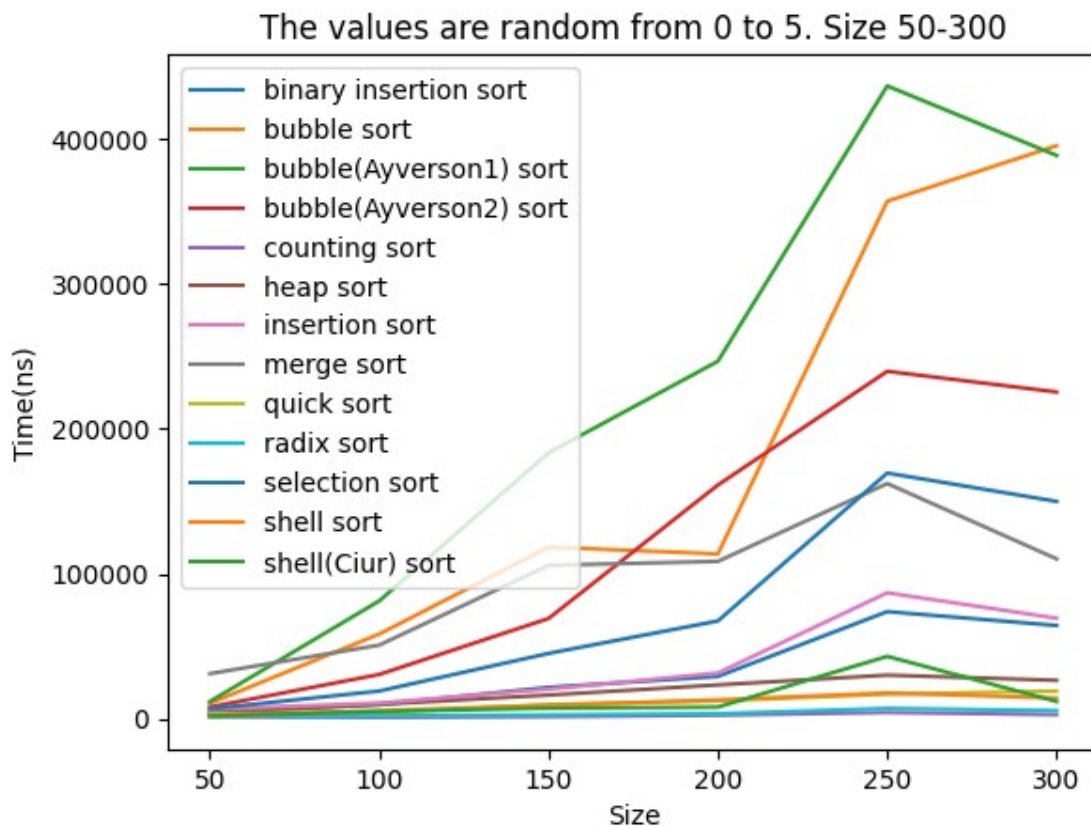


Результаты замеров в графиках и выводы:

График, показывающий производительность разных сортировок по времени. График описывает массивы размером от 50 до 300 с случайными элементами [0;5]

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=';')
filtered_df = df[(df['Type'] == 'basic mass') & (df['Size'] <= 300)]
grouped_df = filtered_df.groupby('Sort')
for name, group in grouped_df:
    plt.plot(group['Size'], group['Time(ns)'], label=str(name))

plt.xlabel('Size')
plt.ylabel('Time(ns)')
plt.title('The values are random from 0 to 5. Size 50-300')
plt.legend()
plt.show()
```



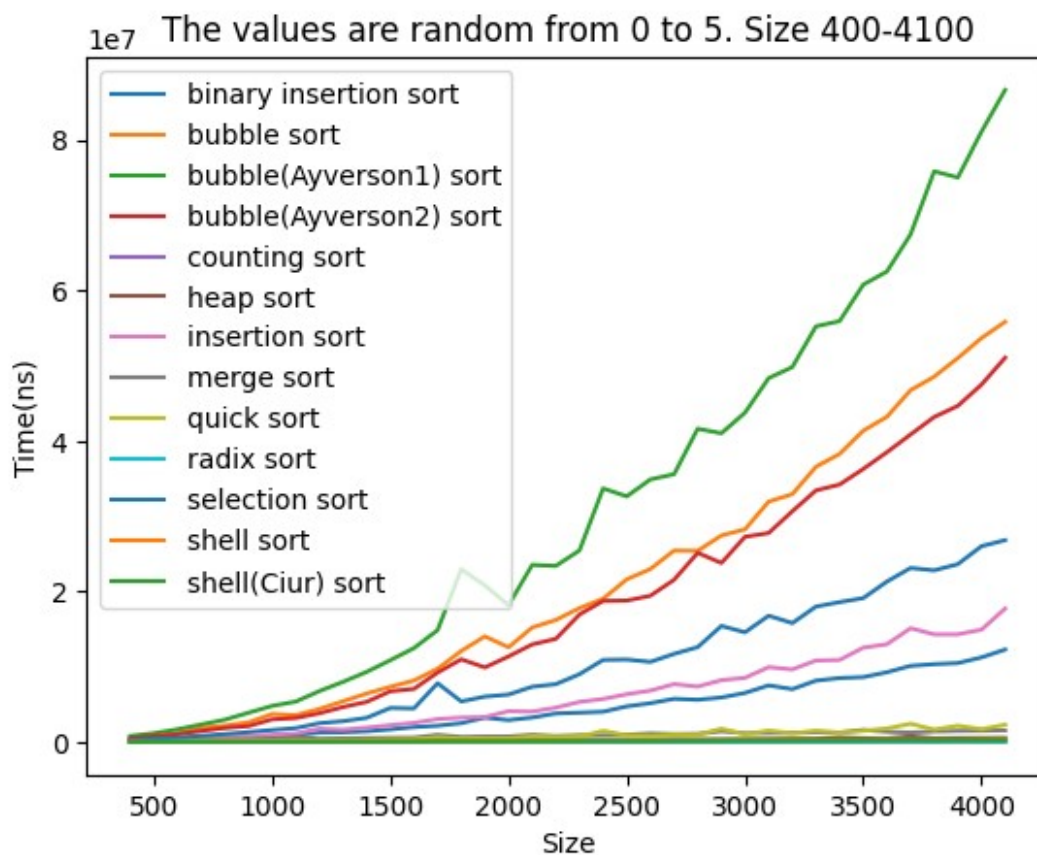
График, показывающий производительность разных сортировок по времени. График описывает массивы размером от 400 до 4100 с случайными элементами [0;5]

```

import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=';')
filtered_df = df[(df['Type'] == 'basic mass') & (df['Size'] >= 400)]
grouped_df = filtered_df.groupby('Sort')
for name, group in grouped_df:
    plt.plot(group['Size'], group['Time(ns)'], label=str(name))

plt.xlabel('Size')
plt.ylabel('Time(ns)')
plt.title('The values are random from 0 to 5. Size 400-4100')
plt.legend()
plt.show()

```



И данном графике мы можем заметить выбросы у всех сортировок, так как происходят и параллельные процессы в компьютере, и убавляет производительность для подсчёта сортировок.

График, показывающий производительность разных сортировок по времени. График описывает массивы размером от 50 до 300 с случайными элементами [0;4000]

```

import matplotlib.pyplot as plt
import pandas as pd

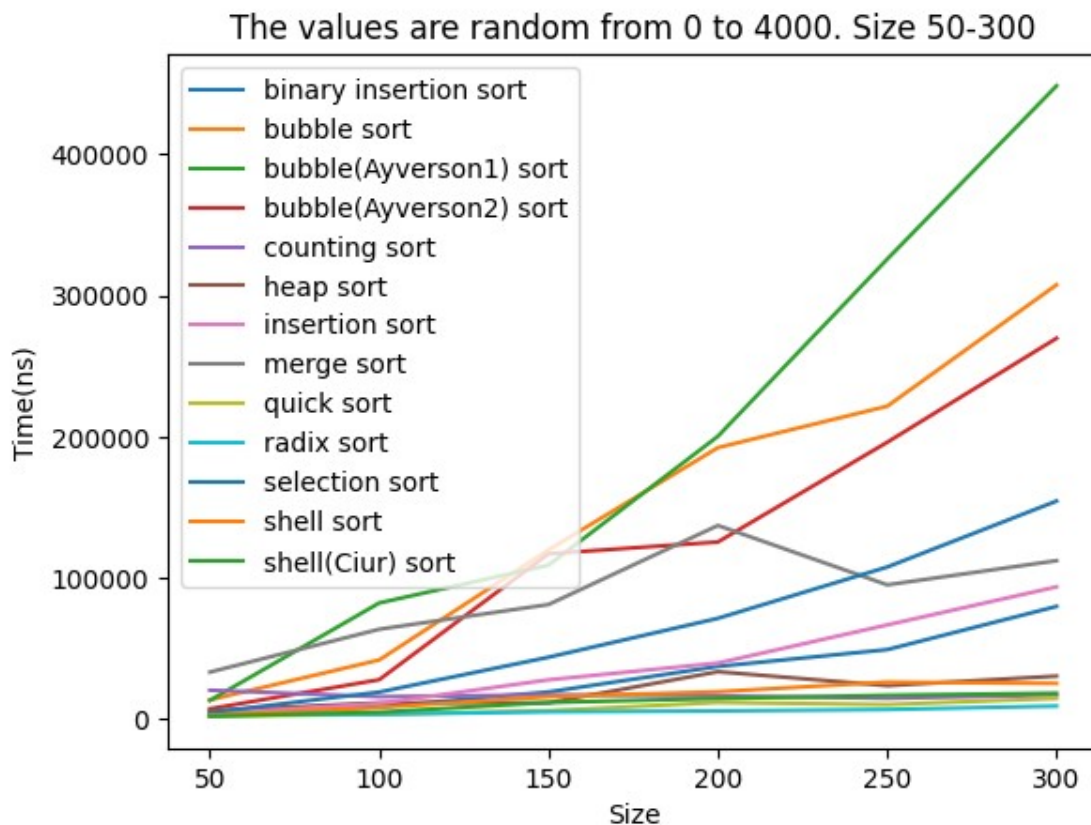
```

```

df = pd.read_csv("../csvTables/table.csv", sep=';')
filtered_df = df[(df['Type'] == 'large mass') & (df['Size'] <= 300)]
grouped_df = filtered_df.groupby('Sort')
for name, group in grouped_df:
    plt.plot(group['Size'], group['Time(ns)'], label=str(name))

plt.xlabel('Size')
plt.ylabel('Time(ns)')
plt.title('The values are random from 0 to 4000. Size 50-300')
plt.legend()
plt.show()

```



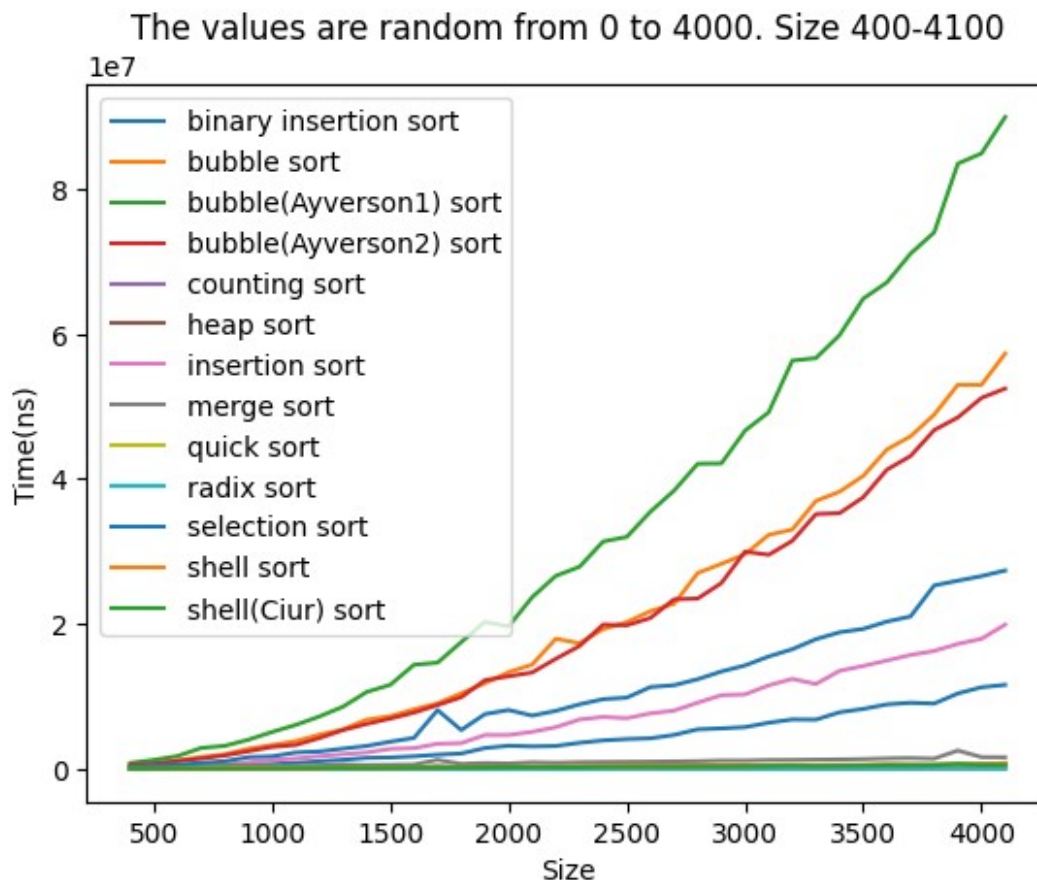
График, показывающий производительность разных сортировок по времени. График описывает массивы размером от 400 до 4100 с случайными элементами [0;4000]

```

import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=';')
filtered_df = df[(df['Type'] == 'large mass') & (df['Size'] >= 400)]
grouped_df = filtered_df.groupby('Sort')
for name, group in grouped_df:
    plt.plot(group['Size'], group['Time(ns)'], label=str(name))

```

```
plt.xlabel('Size')
plt.ylabel('Time(ns)')
plt.title('The values are random from 0 to 4000. Size 400-4100')
plt.legend()
plt.show()
```



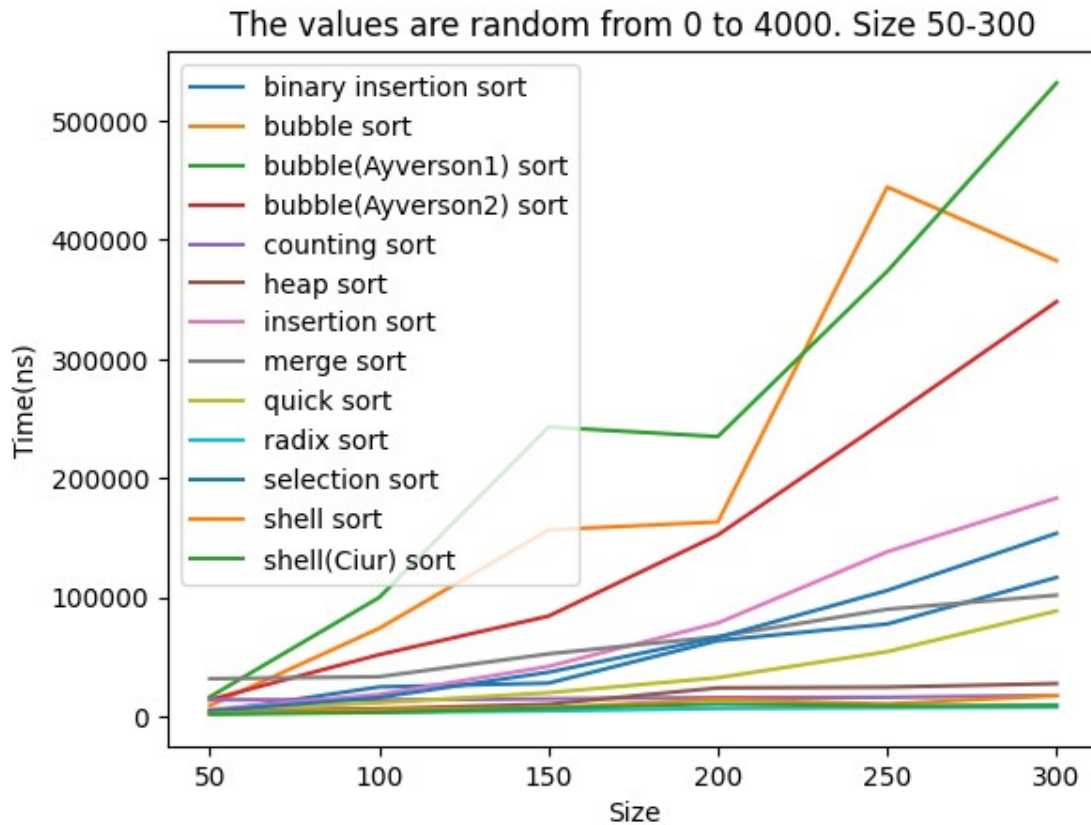
При больших размерах опять можно заметить выбросы.

График, показывающий производительность разных сортировок по времени. График описывает массивы(реверснутые) размером от 50 до 300 с случайными элементами [0;4000]

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=';')
filtered_df = df[(df['Type'] == 'reverse mass') & (df['Size'] <= 300)]
grouped_df = filtered_df.groupby('Sort')
for name, group in grouped_df:
    plt.plot(group['Size'], group['Time(ns)'], label=str(name))

plt.xlabel('Size')
plt.ylabel('Time(ns)')
plt.title('The values are random from 0 to 4000. Size 50-300')
```

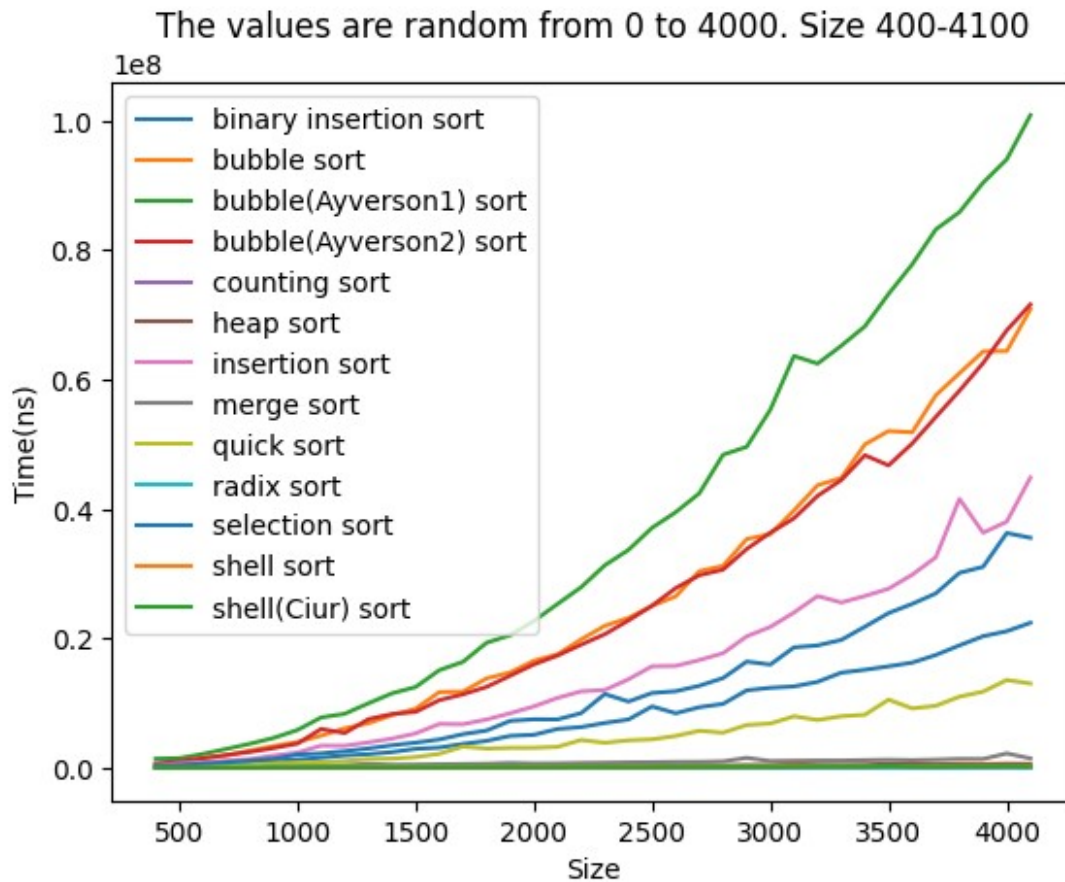
```
plt.legend()
plt.show()
```



График, показывающий производительность разных сортировок по времени. График описывает массивы (реверснутые) размером от 400 до 4100 с случайными элементами [0;4000]

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=';')
filtered_df = df[(df['Type'] == 'reverse mass') & (df['Size'] >= 400)]
grouped_df = filtered_df.groupby('Sort')
for name, group in grouped_df:
    plt.plot(group['Size'], group['Time(ns)'], label=str(name))

plt.xlabel('Size')
plt.ylabel('Time(ns)')
plt.title('The values are random from 0 to 4000. Size 400-4100')
plt.legend()
plt.show()
```

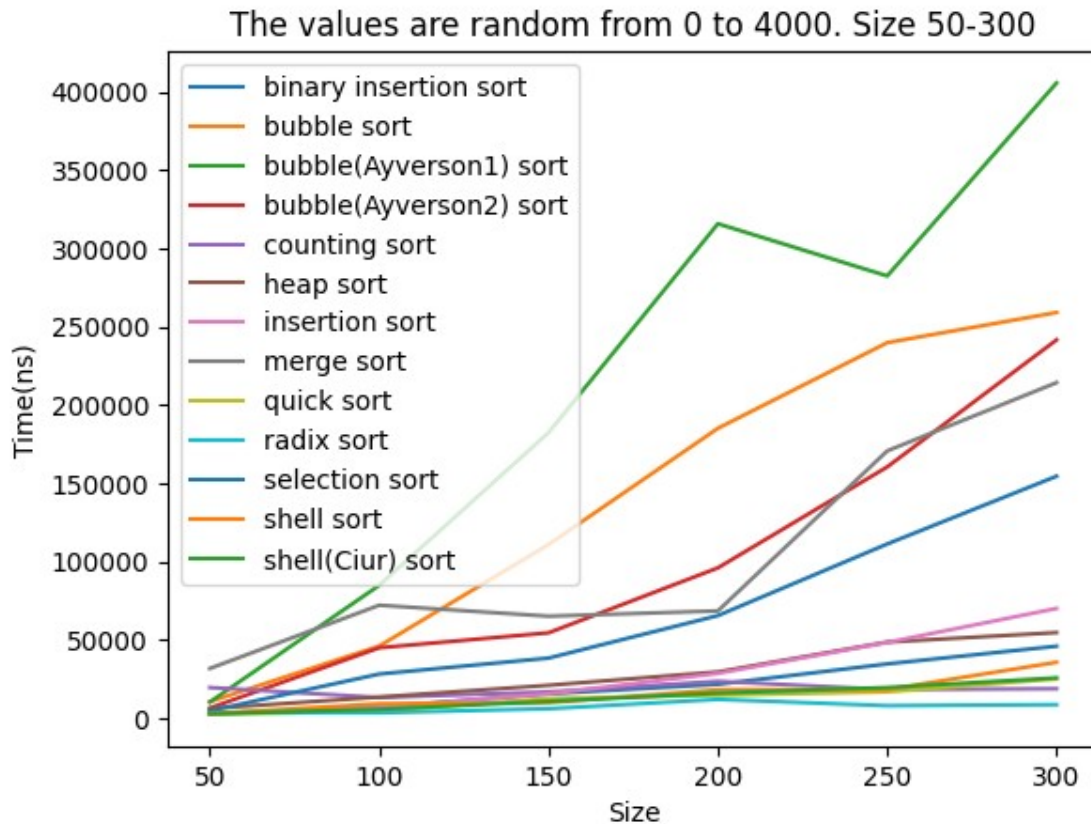


График, показывающий производительность разных сортировок по времени. График описывает массивы (частично отсортированные) размером от 50 до 300 с случайными элементами [0;4000]

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=';')
filtered_df = df[(df['Type'] == 'half sort mass') & (df['Size'] <= 300)]
grouped_df = filtered_df.groupby('Sort')
for name, group in grouped_df:
    plt.plot(group['Size'], group['Time(ns)'], label=str(name))

plt.xlabel('Size')
plt.ylabel('Time(ns)')
plt.title('The values are random from 0 to 4000. Size 50-300')
plt.legend()
plt.show()
```

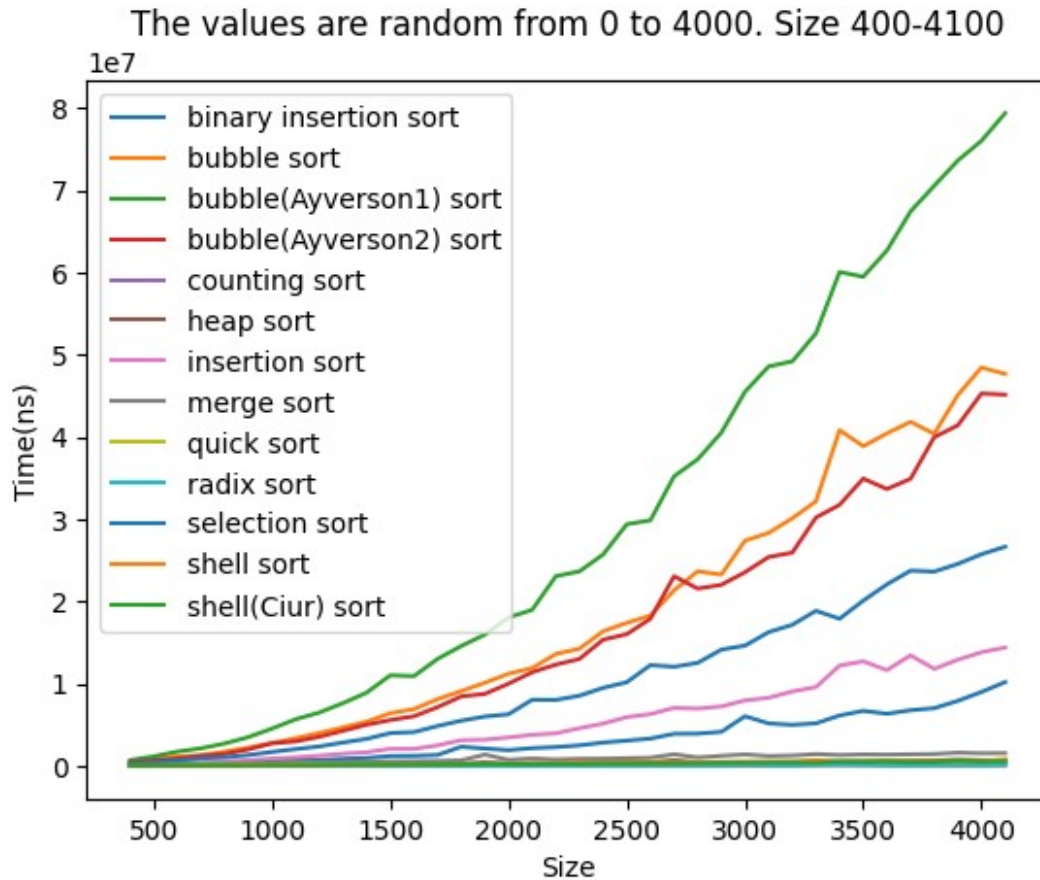




График, показывающий производительность разных сортировок по времени. График описывает массивы (частично отсортированные) размером от 400 до 4100 с случайными элементами [0;4000]

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=';')
filtered_df = df[(df['Type'] == 'half sort mass') & (df['Size'] >= 400)]
grouped_df = filtered_df.groupby('Sort')
for name, group in grouped_df:
    plt.plot(group['Size'], group['Time(ns)'], label=str(name))

plt.xlabel('Size')
plt.ylabel('Time(ns)')
plt.title('The values are random from 0 to 4000. Size 400-4100')
plt.legend()
plt.show()
```

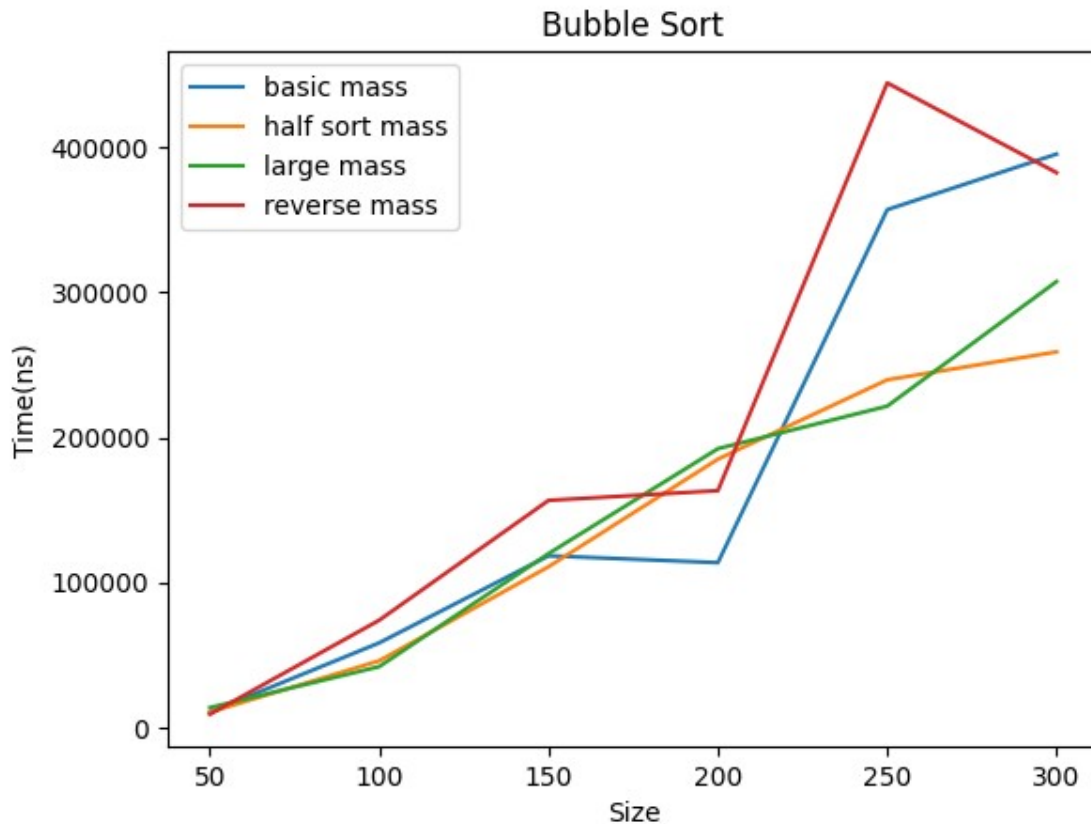


В следующем блоке будут идти графики сортировок(отдельно) и их усовершенствования(если они есть)

### ПУЗЫРЬКОВАЯ СОРТИРОВКА (РАЗМЕР МАЛЕНЬКИЙ)

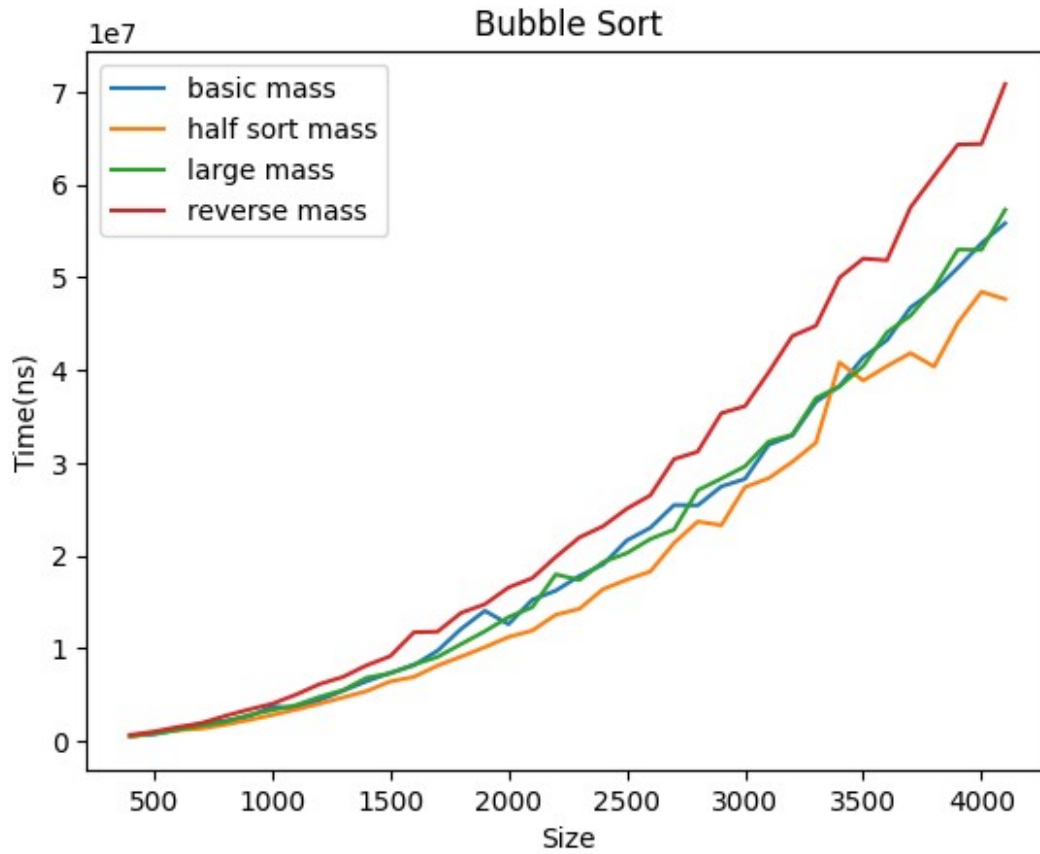
```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "bubble sort") & (df["Size"] <= 300)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Bubble Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```





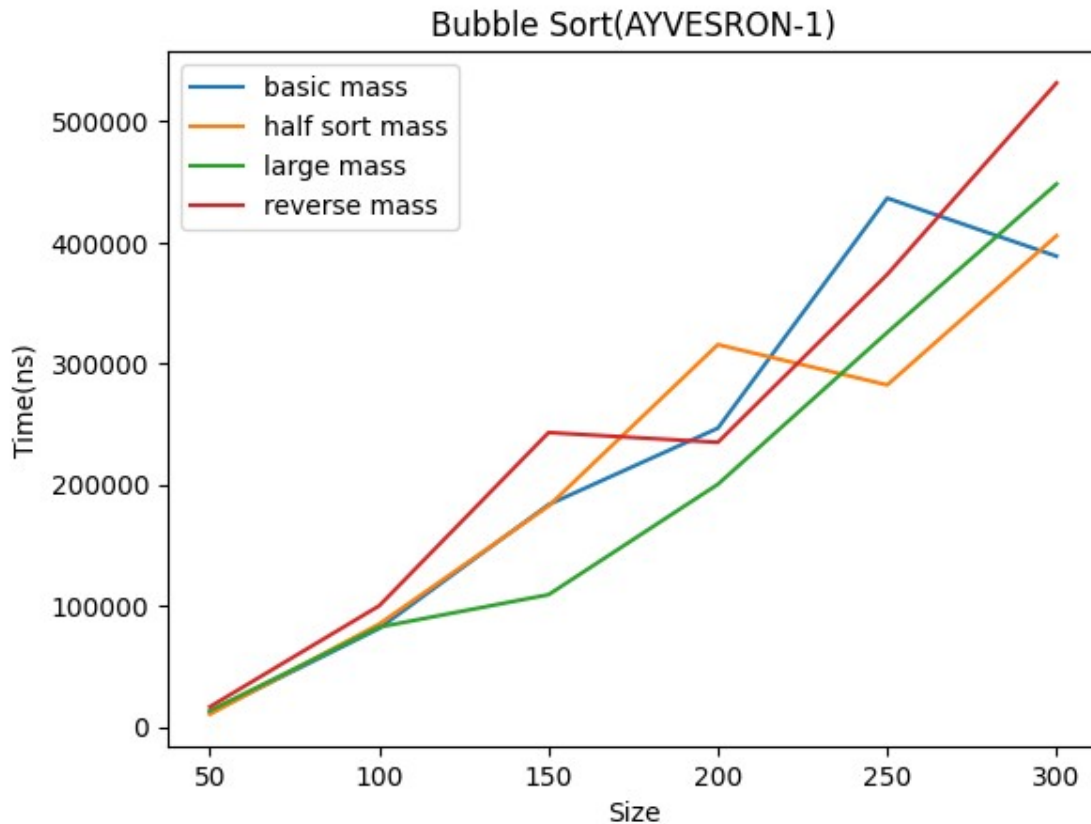
## ПУЗЫРЬКОВАЯ СОРТИРОВКА (РАЗМЕР БОЛЬШОЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "bubble sort") & (df["Size"] >= 400)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Bubble Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



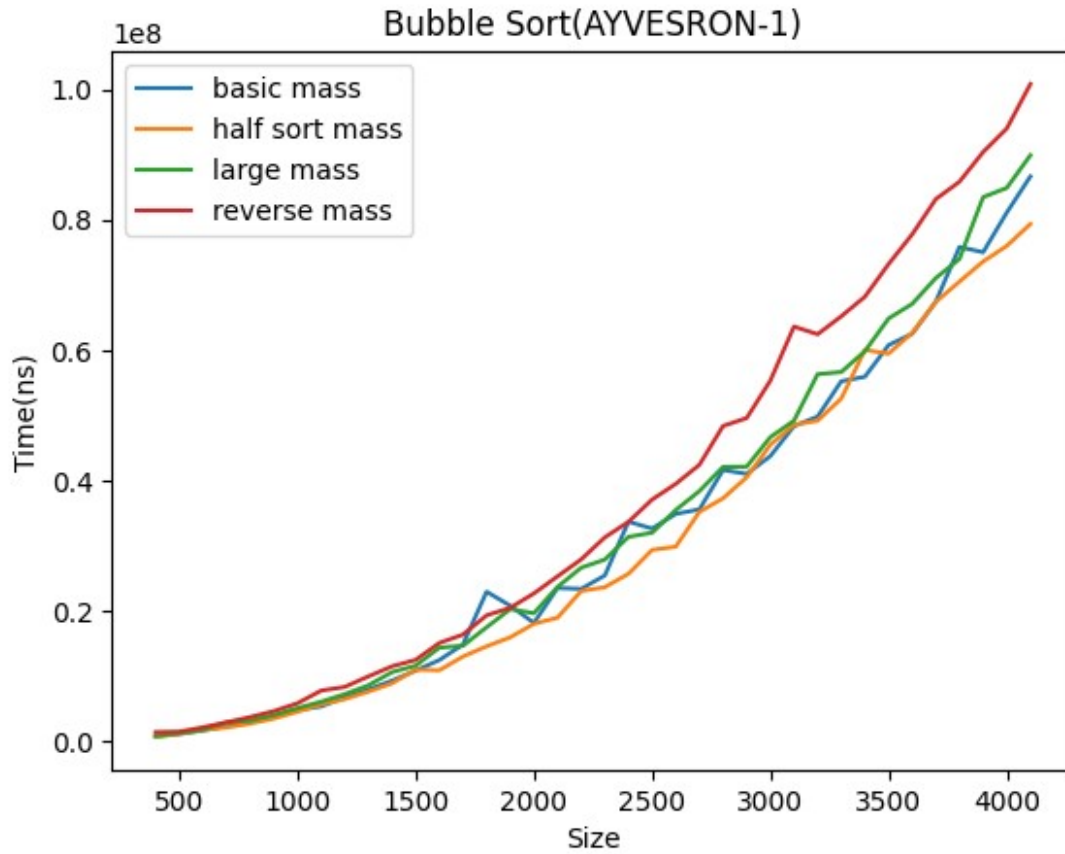
ПУЗЫРЬКОВАЯ(АЙВЕРСОН-1) (РАЗМЕР МАЛЕНЬКИЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "bubble(Ayverson1) sort") & (df["Size"] <= 300)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Bubble Sort(AYVESRON-1)")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



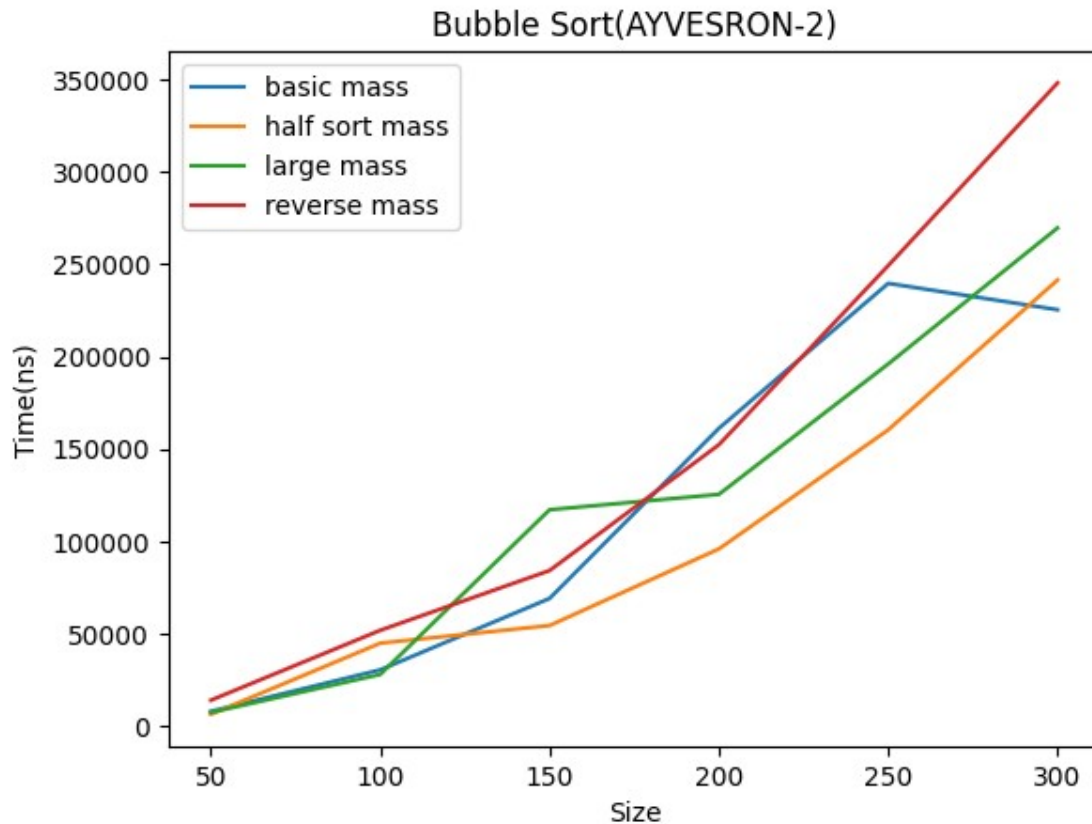
ПУЗЫРЬКОВАЯ(АЙВЕРСОН-1) (РАЗМЕР БОЛЬШОЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "bubble(Ayverson1) sort") & (df["Size"] >=
400)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Bubble Sort(AYVESRON-1)")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



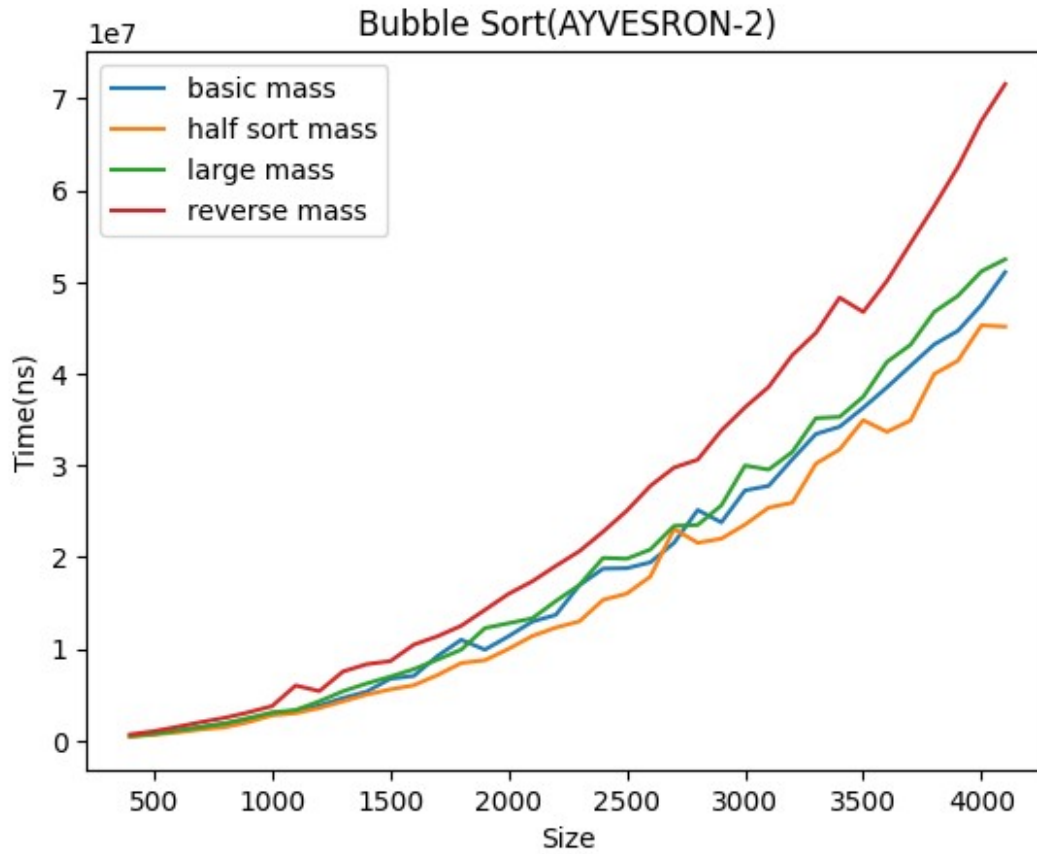
ПУЗЫРЬКОВАЯ(АЙВЕРСОН-2) (РАЗМЕР МАЛЕНЬКИЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "bubble(Ayverson2) sort") & (df["Size"] <= 300)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Bubble Sort(AYVESRON-2)")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



ПУЗЫРЬКОВАЯ(АЙВЕРСОН-2) (РАЗМЕР БОЛЬШОЙ)

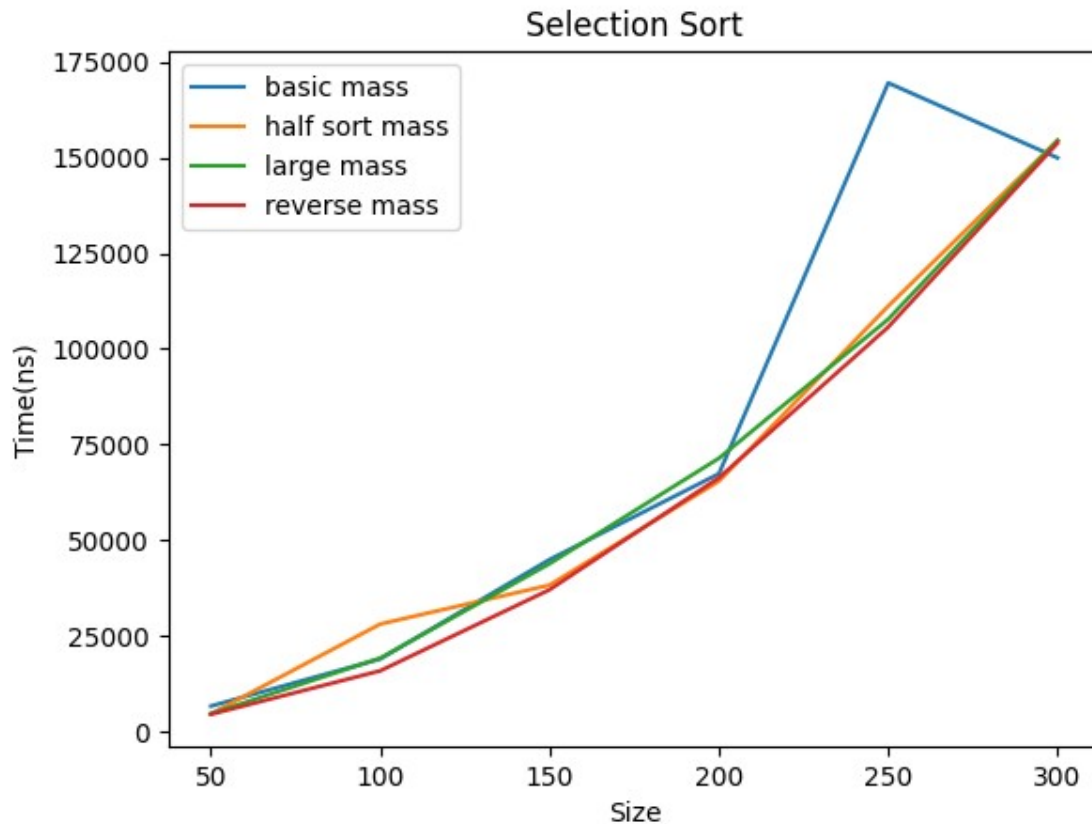
```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "bubble(Ayverson2) sort") & (df["Size"] >=
400)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Bubble Sort(AYVESRON-2)")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



ВЫБОРОМ (РАЗМЕР МАЛЕНЬКИЙ)

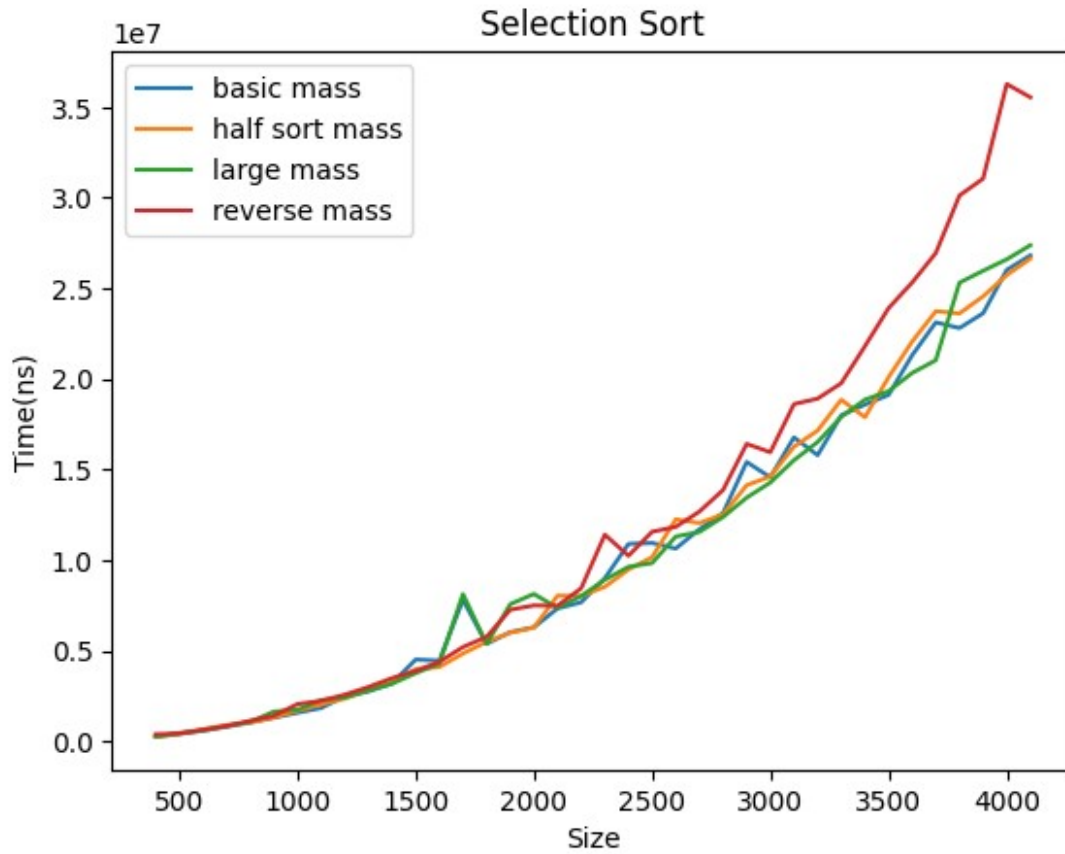
```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "selection sort") & (df["Size"] <= 300)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Selection Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```





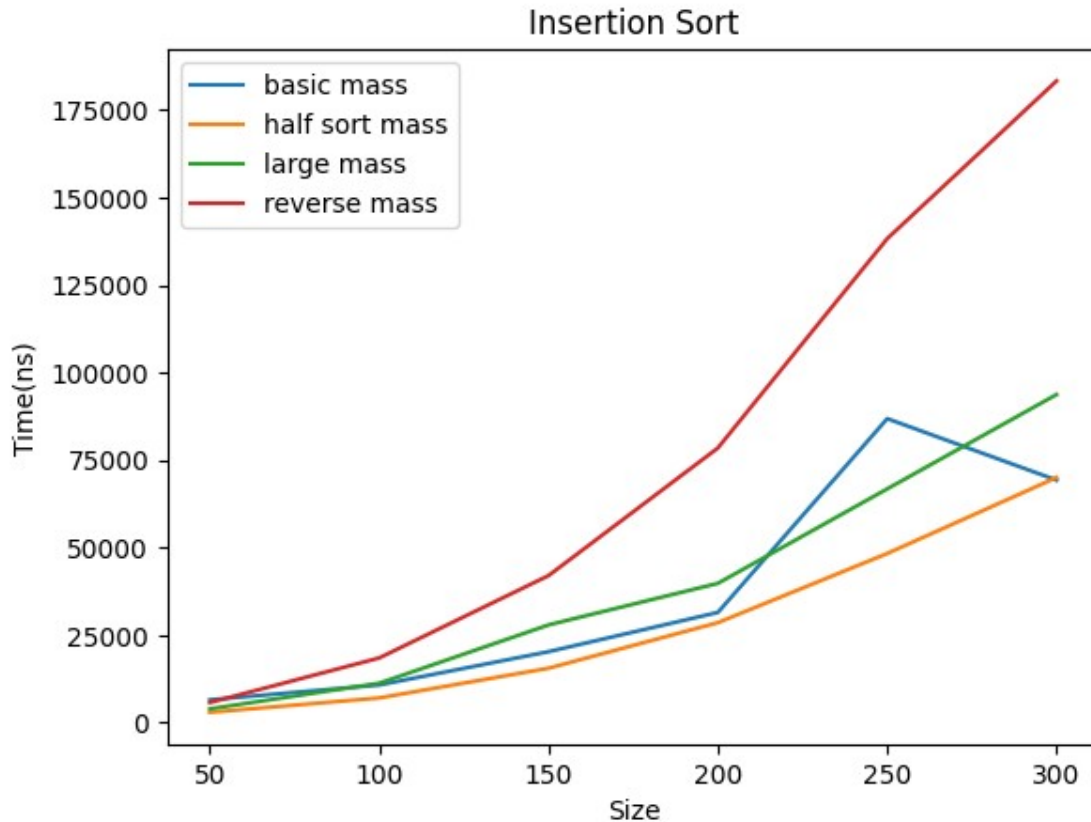
ВЫБОРОМ (РАЗМЕР БОЛЬШОЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "selection sort") & (df["Size"] >= 400)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Selection Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



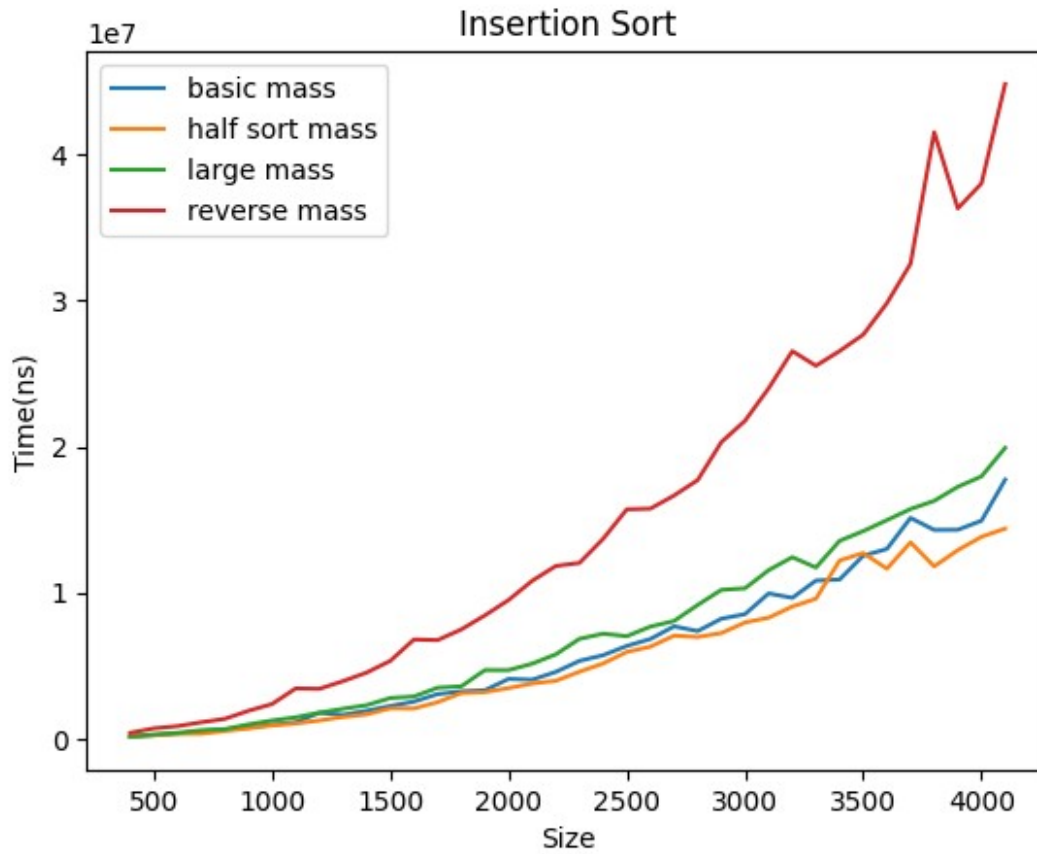
ПРОСТЫМИ ВСТАВКАМИ (РАЗМЕР МАЛЕНЬКИЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "insertion sort") & (df["Size"] <= 300)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Insertion Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



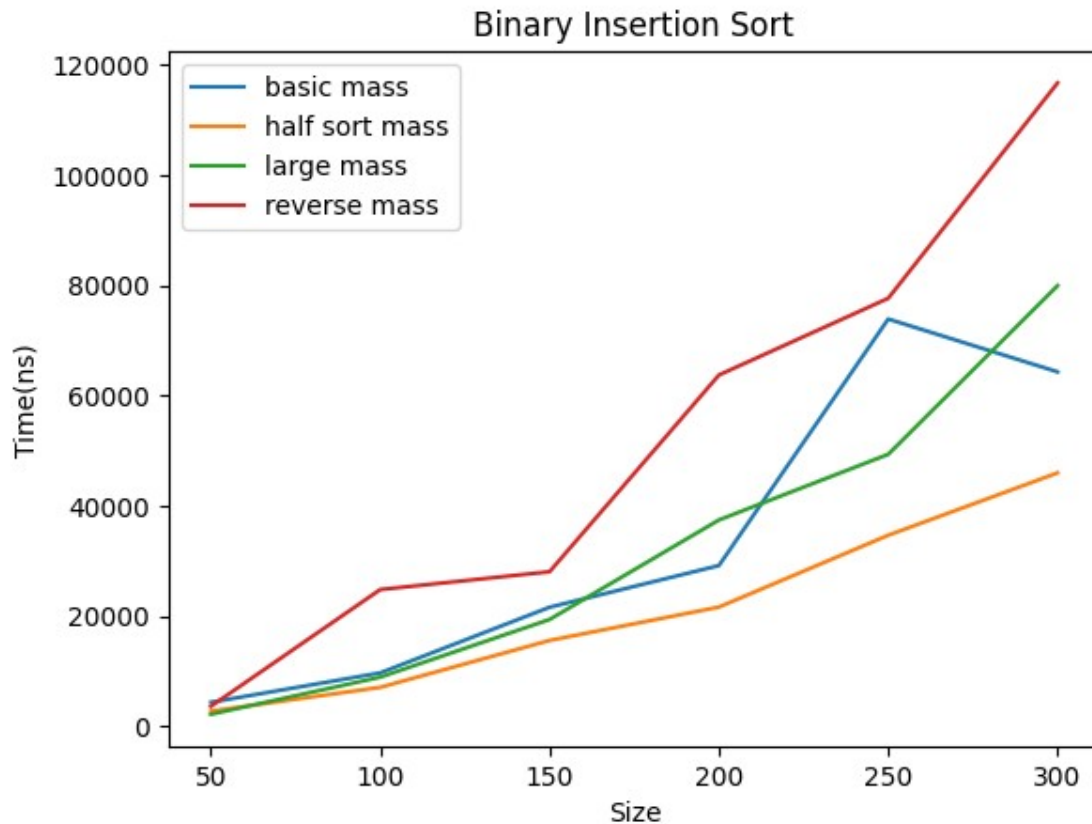
ПРОСТЫМИ ВСТАВКАМИ (РАЗМЕР БОЛЬШОЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "insertion sort") & (df["Size"] >= 400)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Insertion Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



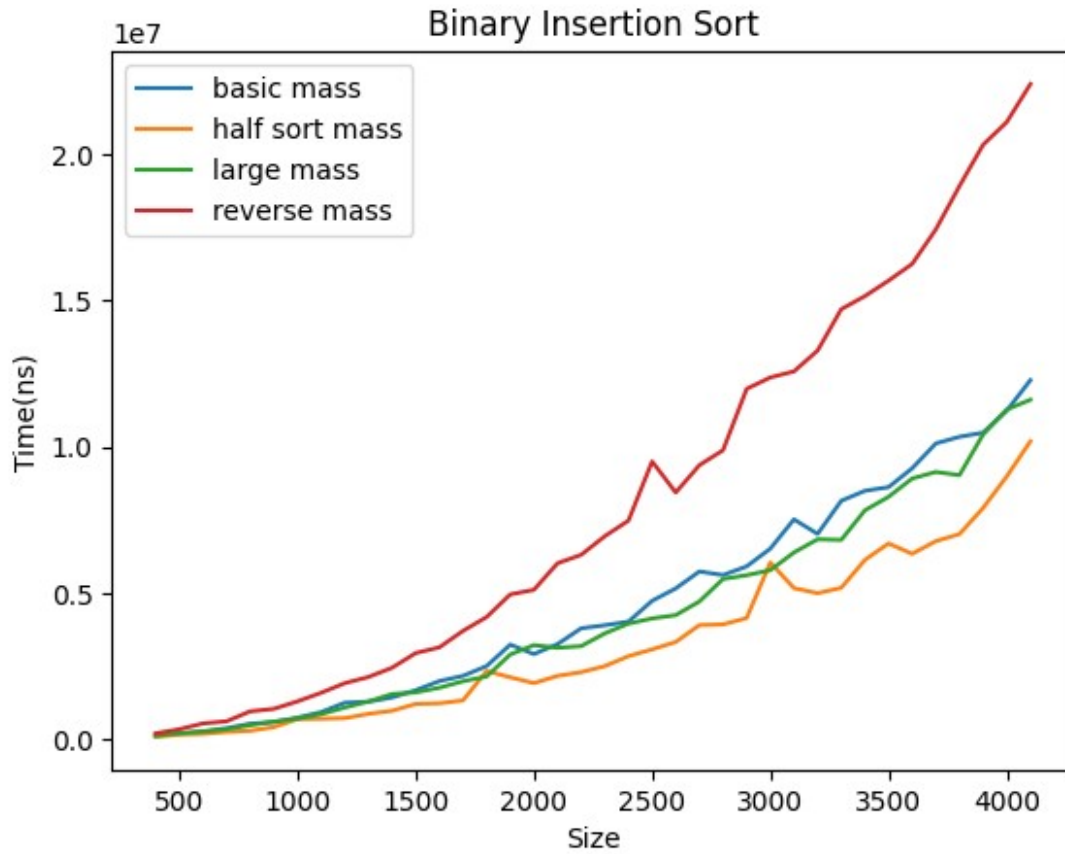
БИНАРНЫМИ ВСТАВКАМИ (РАЗМЕР МАЛЕНЬКИЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "binary insertion sort") & (df["Size"] <= 300)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Binary Insertion Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



БИНАРНЫМИ ВСТАВКАМИ (РАЗМЕР БОЛЬШОЙ)

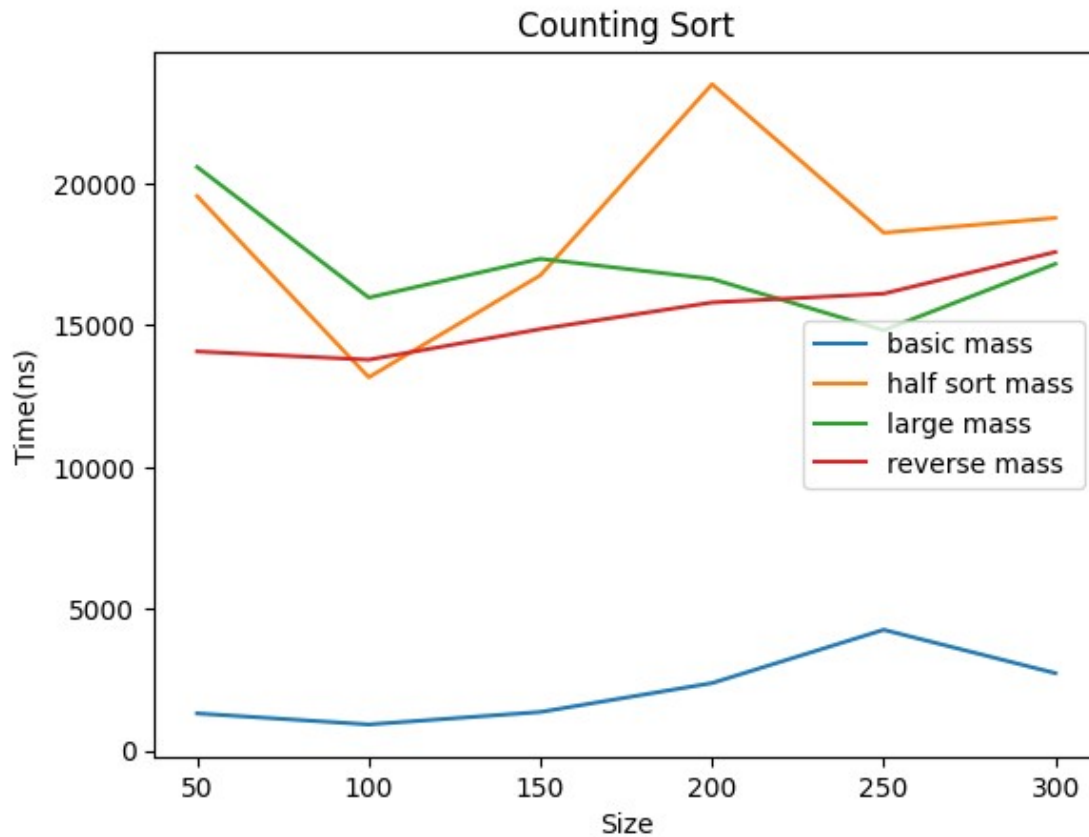
```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "binary insertion sort") & (df["Size"] >= 400)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Binary Insertion Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



ПОДСЧЁТОМ(УСТОЙЧИВАЯ) (РАЗМЕР МАЛЕНЬКИЙ)

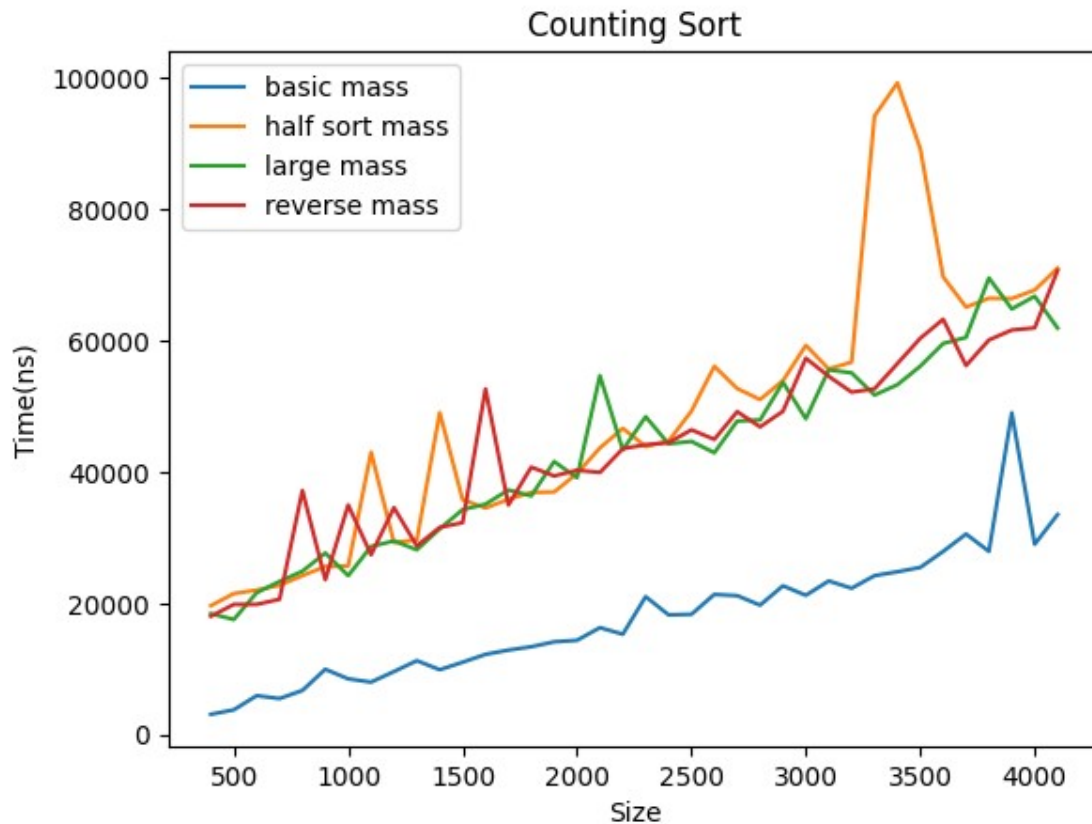
```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "counting sort") & (df["Size"] <= 300)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Counting Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```





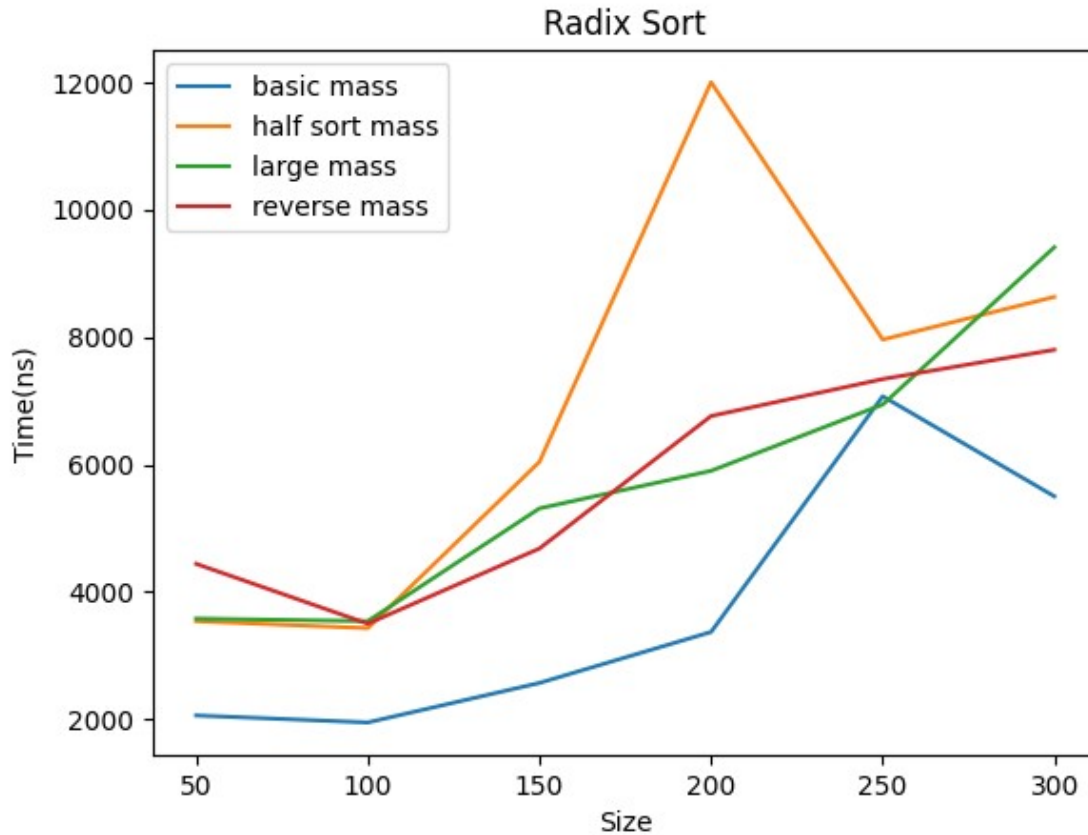
ПОДСЧЁТОМ(УСТОЙЧИВАЯ) (РАЗМЕР БОЛЬШОЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "counting sort") & (df["Size"] >= 400)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Counting Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



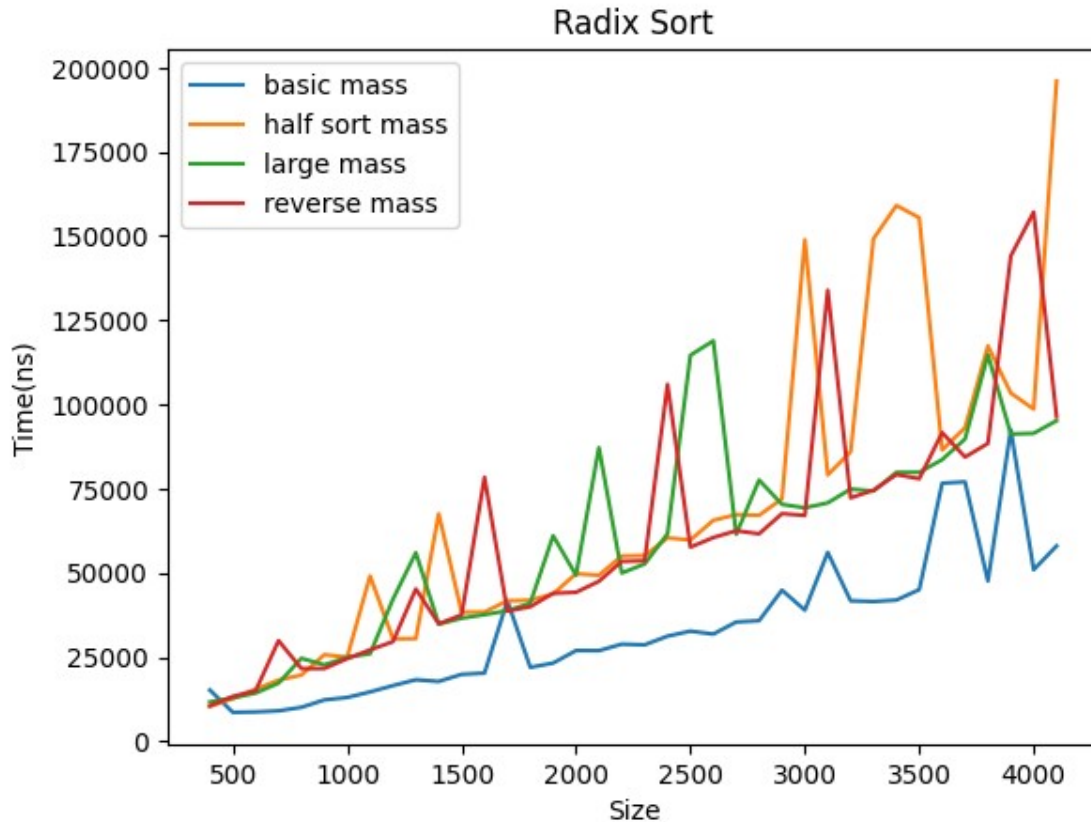
ЦИФРОВАЯ (РАЗМЕР МАЛЕНЬКИЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "radix sort") & (df["Size"] <= 300)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Radix Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



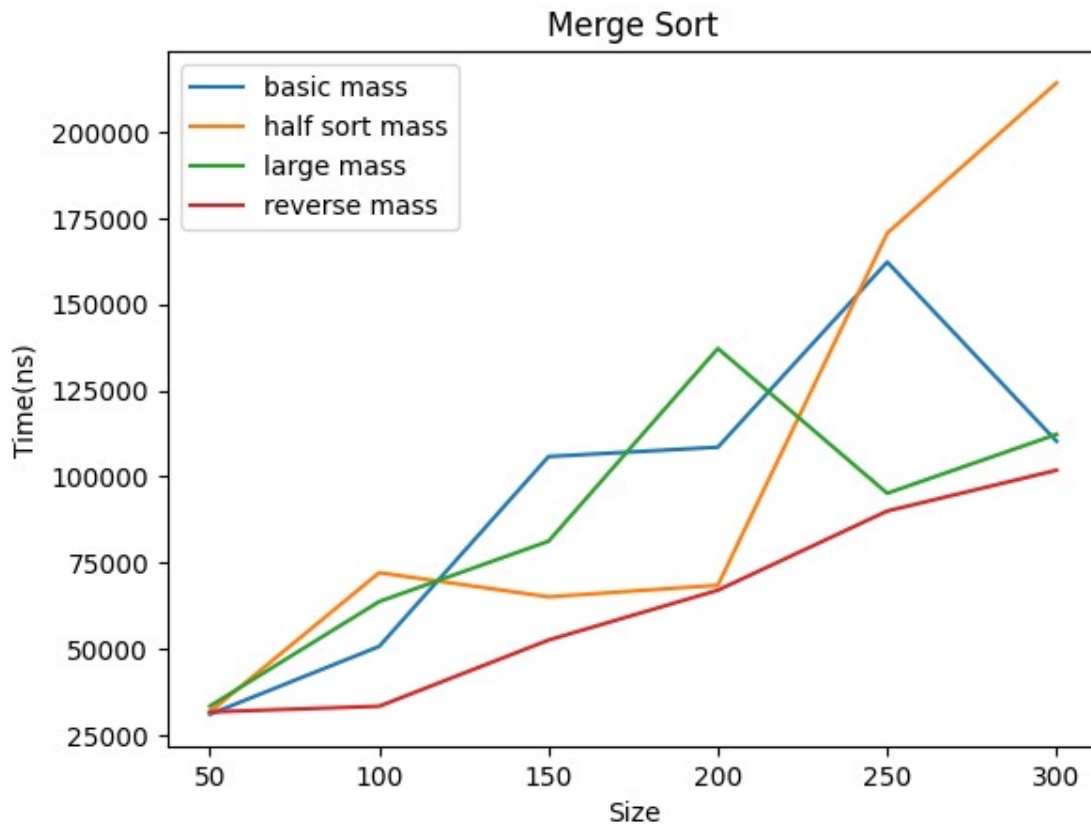
ЦИФРОВАЯ (РАЗМЕР БОЛЬШОЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "radix sort") & (df["Size"] >= 400)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Radix Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



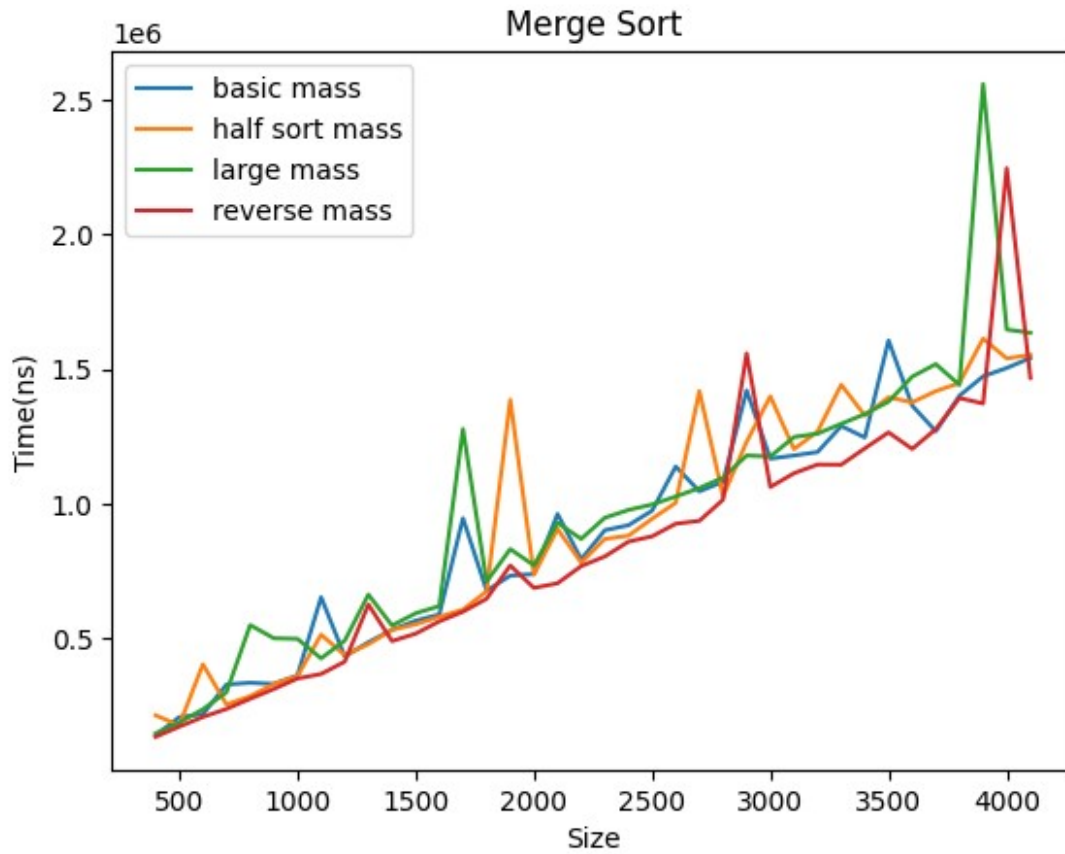
СЛИЯНИЕМ (РАЗМЕР МАЛЕНЬКИЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "merge sort") & (df["Size"] <= 300)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Merge Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



СЛИЯНИЕМ (РАЗМЕР БОЛЬШОЙ)

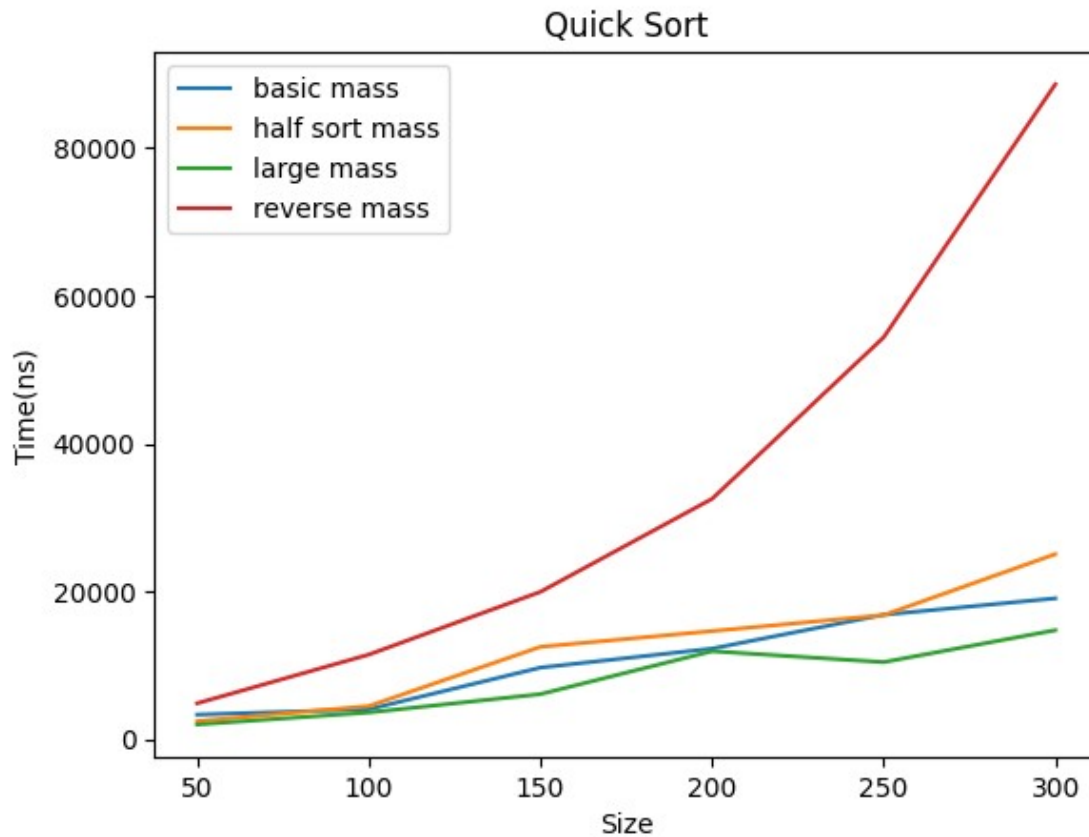
```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "merge sort") & (df["Size"] >= 400)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Merge Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



БЫСТРАЯ (РАЗМЕР МАЛЕНЬКИЙ)

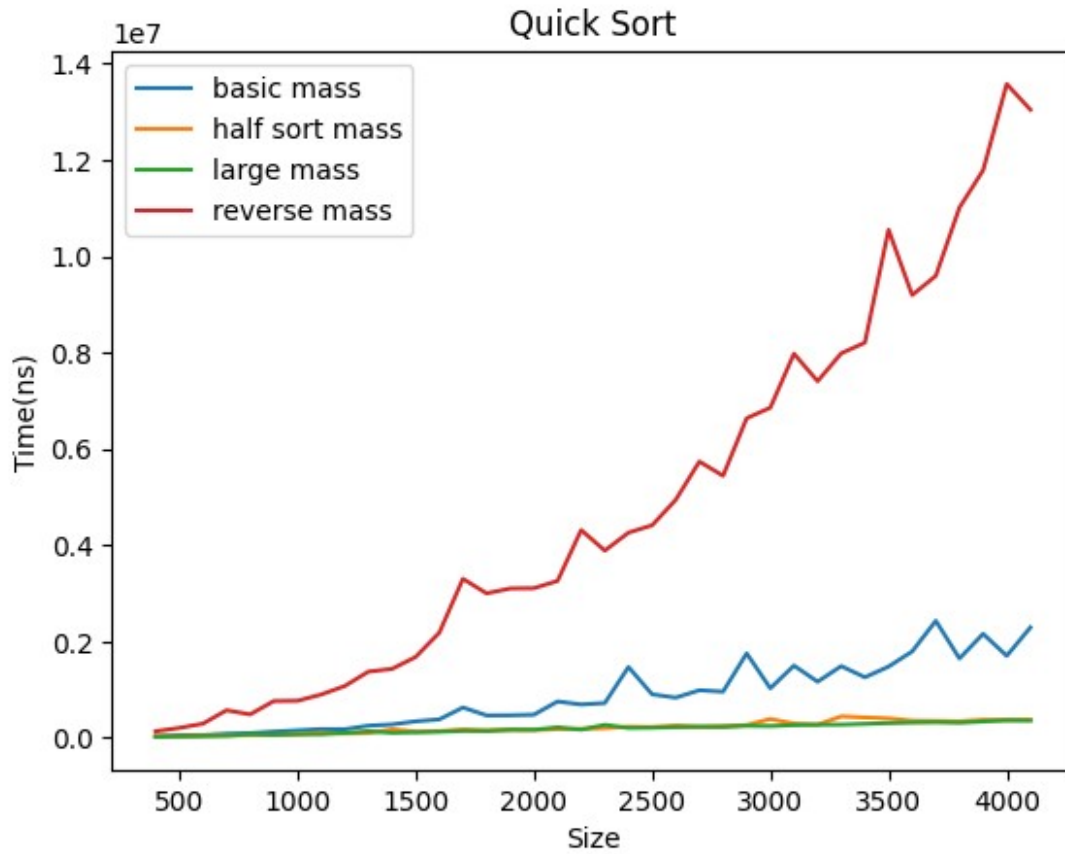
```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "quick sort") & (df["Size"] <= 300)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Quick Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```





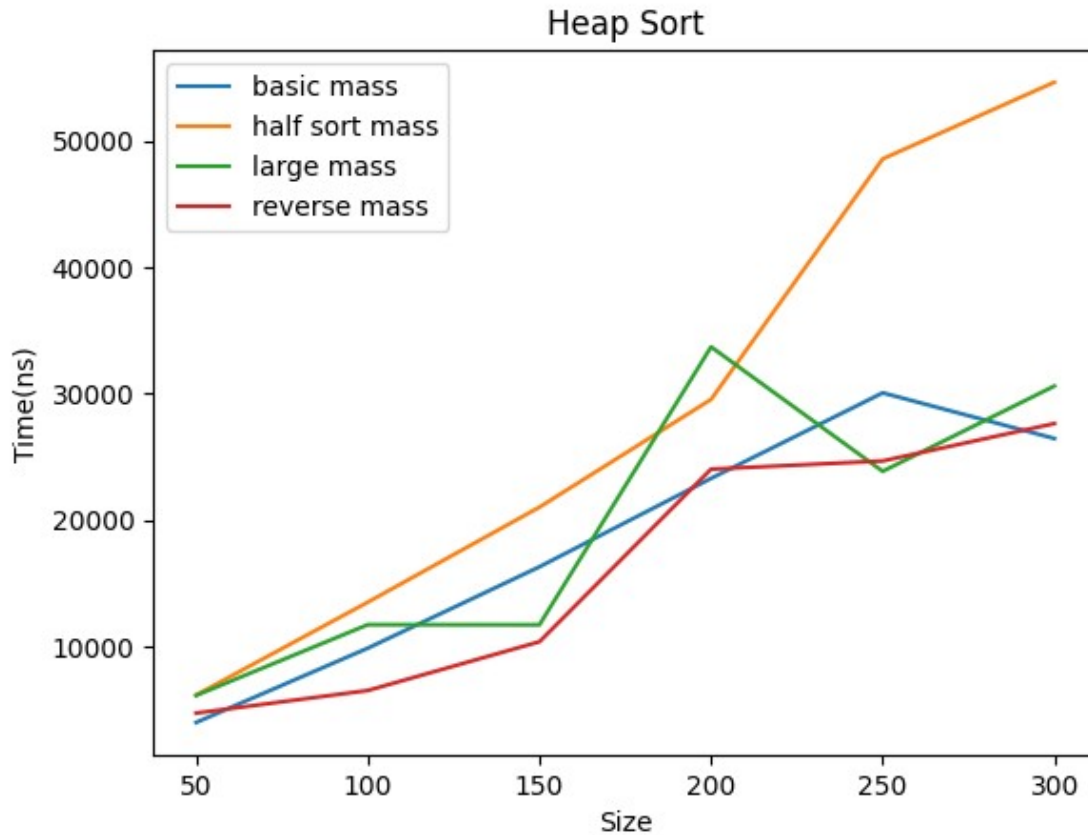
БЫСТРАЯ (РАЗМЕР БОЛЬШОЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "quick sort") & (df["Size"] >= 400)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Quick Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



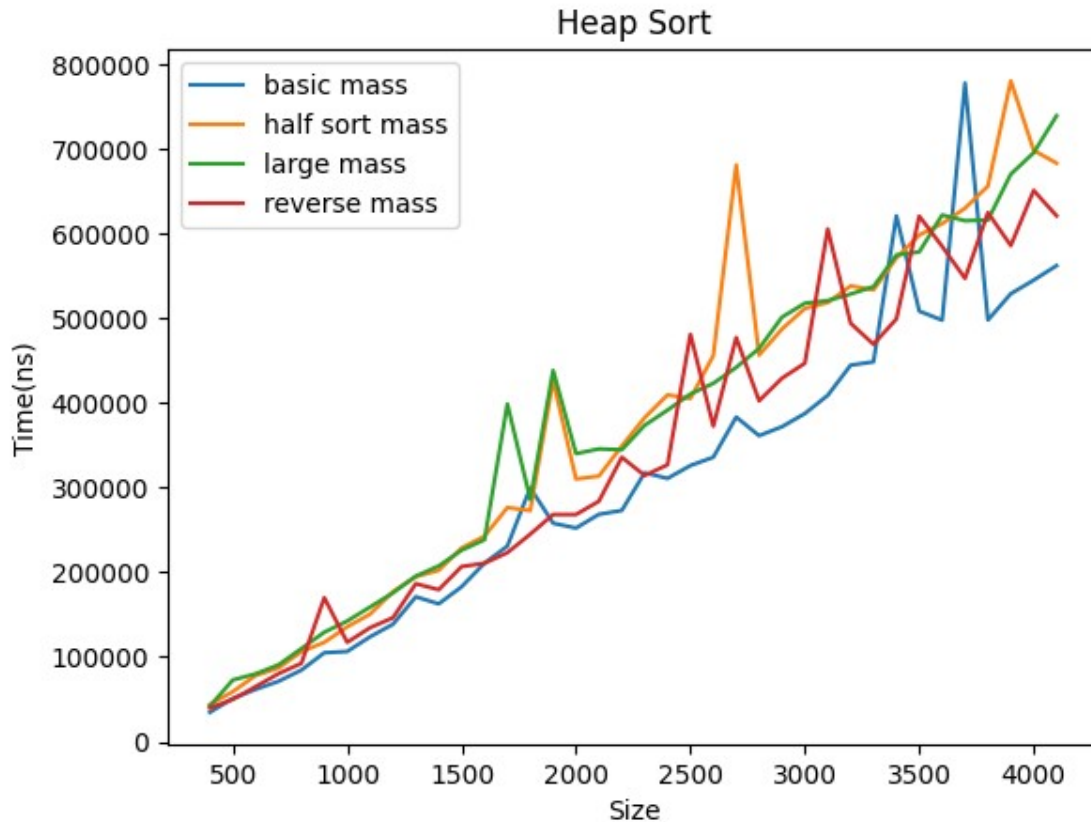
ПИРАМИДАЛЬНАЯ (РАЗМЕР МАЛЕНЬКИЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "heap sort") & (df["Size"] <= 300)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Heap Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



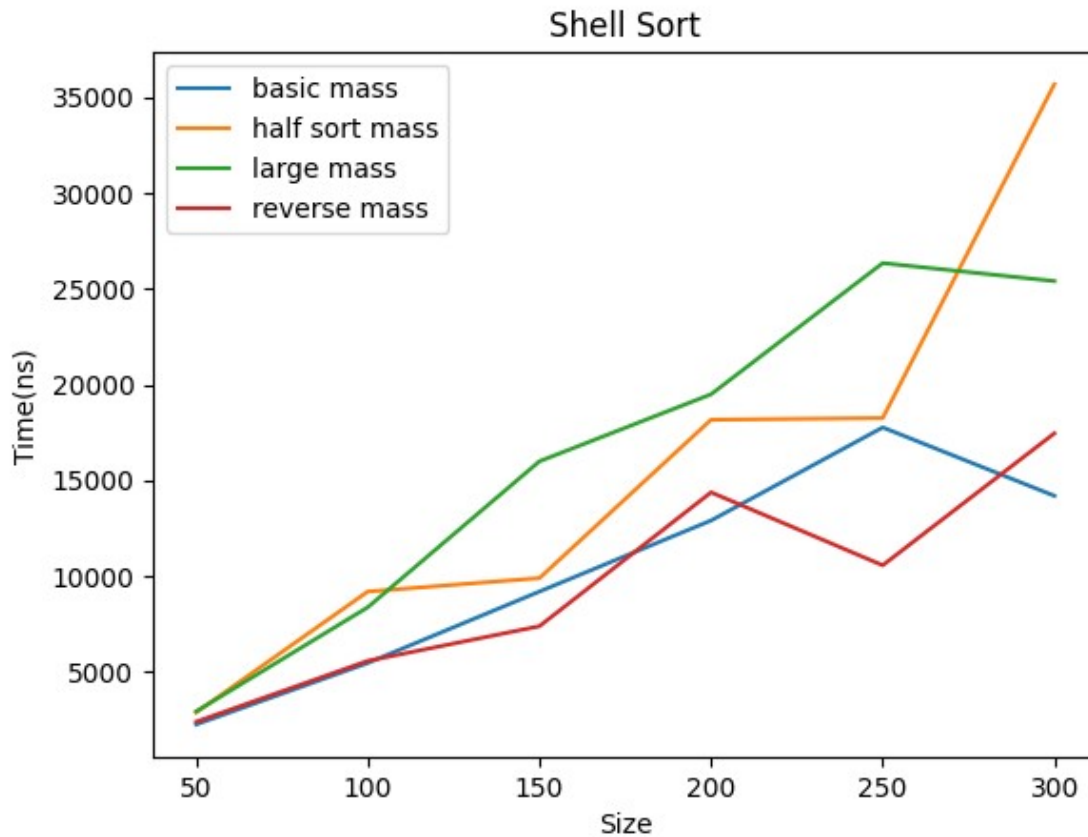
ПИРАМИДАЛЬНАЯ (РАЗМЕР БОЛЬШОЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "heap sort") & (df["Size"] >= 400)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Heap Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



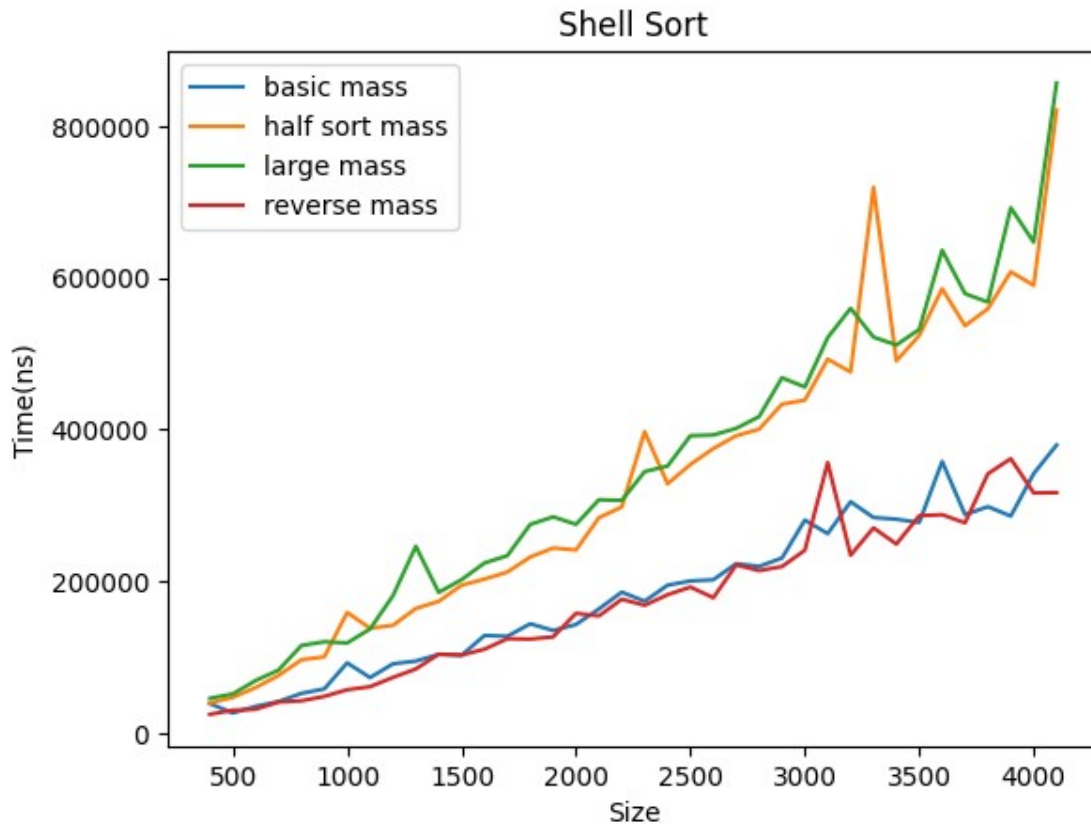
ШЕЛЛА(ПОСЛЕДОВАТЕЛЬНОСТЬ ШЕЛЛА) (РАЗМЕР МАЛЕНЬКИЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "shell sort") & (df["Size"] <= 300)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Shell Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



ШЕЛЛА(ПОСЛЕДОВАТЕЛЬНОСТЬ ШЕЛЛА) (РАЗМЕР БОЛЬШОЙ)

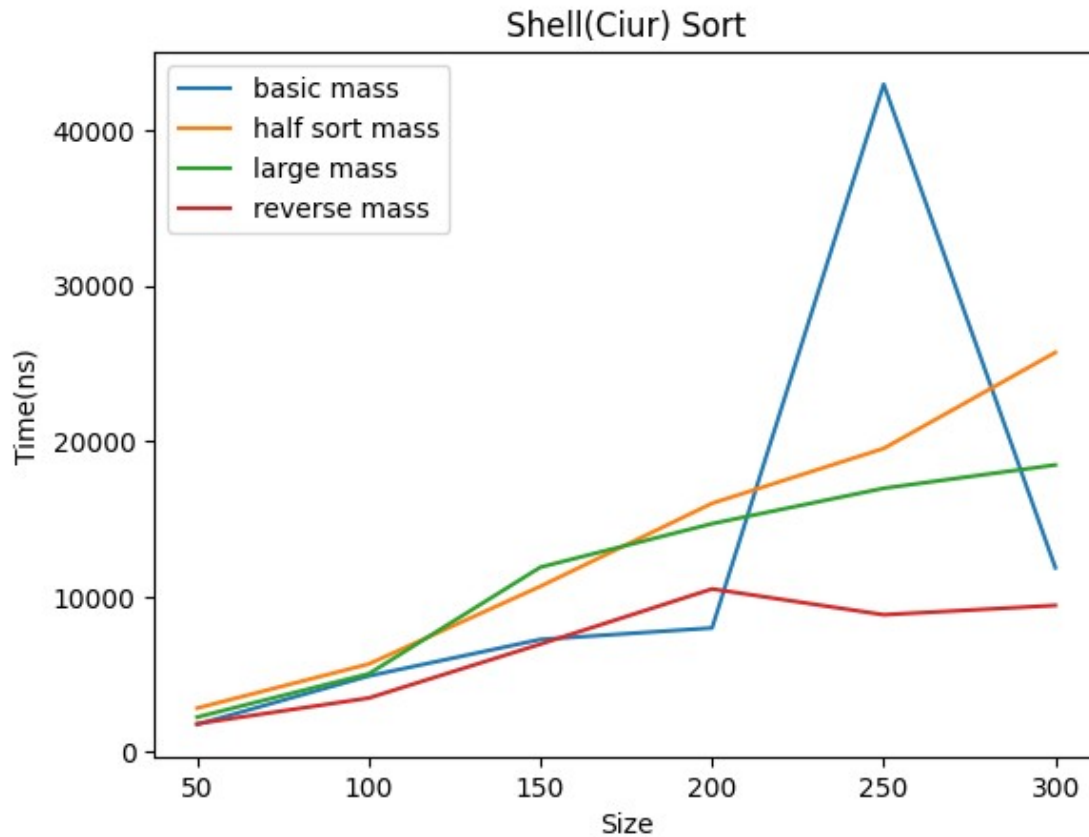
```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "shell sort") & (df["Size"] >= 400)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Shell Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```



ШЕЛЛА(ПОСЛЕДОВАТЕЛЬНОСТЬ ЦИУРА) (РАЗМЕР МАЛЕНЬКИЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "shell(Ciur) sort") & (df["Size"] <= 300)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Shell(Ciur) Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```





ШЕЛЛА(ПОСЛЕДОВАТЕЛЬНОСТЬ ЦИУРА) (РАЗМЕР БОЛЬШОЙ)

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("../csvTables/table.csv", sep=";")
df = df[(df["Sort"] == "shell(Ciur) sort") & (df["Size"] >= 400)]
grouped = df.groupby("Type")
fig, ax = plt.subplots()
for name, group in grouped:
    ax.plot(group["Size"], group["Time(ns)"], label=name)
ax.set_title("Shell(Ciur) Sort")
ax.set_xlabel("Size")
ax.set_ylabel("Time(ns)")
ax.legend()
plt.show()
```

Shell(Ciur) Sort

