Были получены результаты по работе программы, созданы 10 таблиц: по две на каждое изменение шаблонов (по количеству измененных символов на '?') - измеренное время работы и измеренное количество посимвольных сравнений. Шаблоны были сгенерированы для каждого текста, и алгоритмы прогонялись по одинаковым шаблонам. Изменения шаблона происходили так, что заданее генерировались 4 различных индекса строки, на которых будет производиться замена символа на '?'. И постепенно, по одному индексу, шаблоны меняются, и алгоритмы прогоняются по новым шаблонам.

На графиках можно увидеть, что наивный поиск во всех случаях имеет меньшее количество посимвольных сравнений, чем алгоритмы КМП, а также в среднем на него тратится меньше вычислительного времени.

С ростом размера шаблона количество посимвольных сравнений у наивного поиска уменьшается, в то время как у алгоритмов КМП увеличивается. Это связано с тем, что алгоритмам КМП необходимо считать грани и уточненные грани.

На большом бинарном тексте алгоритмы КМП тратят меньше времени на работу, чем наивный поиск.

Между собой стандартный и оптимальный алгоритмы КМП идут примерно вровень, с чуть большим временем выполнения в среднем у оптимального алгоритма.

На небольших шаблонах алгоритмы КМП показывают себя лучше в плане времени выполнения (на бинарных текстах), чем наивный поиск, однако с ростом размера шаблона время работы алгоритмов КМП растет, а время наивного поиска уменьшается.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import itertools
import matplotlib.gridspec as gridspec

algs = ["Naive Comparison", "Standard KMP", "Optimal KMP"]
texts = ["Binary Small Text", "Binary Large Text", "DNA Small Text",
"DNA Large Text"]

Графики для шаблонов без замены символов на '?'
times = pd.read_csv("tables/БПИ213_Кадыкова_0AddsComputingTime.csv",
sep=';')
oper = pd.read_csv("tables/БПИ213_Кадыкова_0AddsOperations.csv",
sep=';')
```

```
#@title Графики измеренного времени по всем алгоритмам для конкретного
текста, где шаблоны оставлены неизменными
#for all algs, computing time table with 0 adds of '?' in samples
gs = gridspec.GridSpec(2, 2)
fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(10, 10))
for curr, grd in zip(texts, itertools.product([0, 1], [0, 1])):
     ax = plt.subplot(gs[grd[0], grd[1]])
     for alg in algs:
          plt.plot(times["Sample sizes"], times[curr + " & " + alg],
label=alg)
     ax.set(xlabel='Sample sizes', ylabel='Nanoseconds',
          title='Time measures of ' + curr)
     plt.legend()
fig.savefig("images/ZeroAddsComputingTime.png")
plt.show()
                                                  1e6 Time measures of Binary Large Text
             Time measures of Binary Small Text
    300000
                      Naive Comparison
                                                                      Naive Comparison
                                               2.1
                      Standard KMP
                                                                      Standard KMP
    280000
                      Optimal KMP
                                                                      Optimal KMP
    260000
                                               2.0
    240000
                                             Nanoseconds
1.8
  Nanoseconds
    220000
    200000
    180000
                                               1.7
    160000
                                               1.6
    140000
                              2000
                                                                1500
                                                                     2000
              500
                        1500
                                   2500
                                                       500
                                                           1000
                                                                           2500
                      Sample sizes
                                                              Sample sizes
              Time measures of DNA Small Text
                                                  <sub>1e6</sub> Time measures of DNA Large Text
               Naive Comparison
                                                       Naive Comparison
    220000
               Standard KMP
                                                       Standard KMP
               Optimal KMP
                                                       Optimal KMP
                                               1.5
    200000
                                             Nanoseconds
1.3
    180000
  Nanoseconds
    160000
    140000
    120000
                                               1.2
    100000
                                               1.1
```

1500

Sample sizes

2000

2500

500

1000

1500

Sample sizes

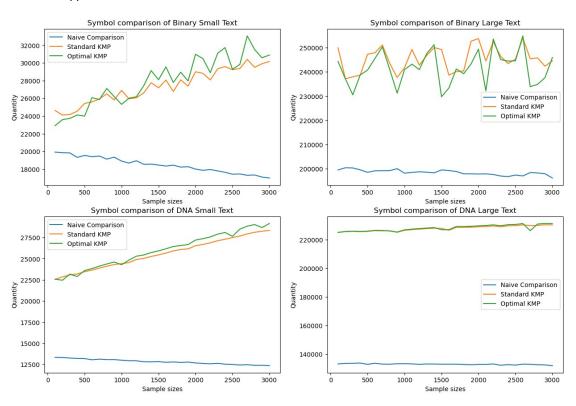
2000

2500

```
#@title Графики количества посимвольных сравнений по всем алгоритмам
для конкретного текста, где шаблоны оставлены неизменными
#for all algs, operations table with 0 adds of '?' in samples
```

```
gs = gridspec.GridSpec(2, 2)
fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(15, 10))
for curr, grd in zip(texts, itertools.product([0, 1], [0, 1])):
    ax = plt.subplot(gs[grd[0], grd[1]])
    for alg in algs:
        plt.plot(oper["Sample sizes"], oper[curr + " & " + alg],
label=alg)
    ax.set(xlabel='Sample sizes', ylabel='Quantity',
        title='Symbol comparison of ' + curr)
    plt.legend()
fig.savefig("images/ZeroAddsOperations.png")
```

plt.show()



Графики для шаблонов с 1 измененным символом

```
times = pd.read csv("tables/БПИ213 Кадыкова 1AddsComputingTime.csv",
sep=';')
oper = pd.read csv("tables/БПИ213 Кадыкова 1AddsOperations.csv",
sep=';')
#@title Графики измеренного времени по всем алгоритмам для конкретного
```

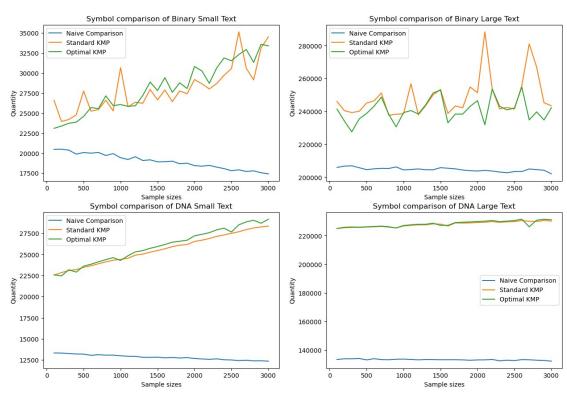
текста, где в шаблоне заменен 1 символ на '?' #for all algs, computing time table with 1 add of '?' in samples

```
gs = gridspec.GridSpec(2, 2)
fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(10, 10))
for curr, grd in zip(texts, itertools.product([0, 1], [0, 1])):
     ax = plt.subplot(gs[grd[0], grd[1]])
     for alg in algs:
           plt.plot(times["Sample sizes"], times[curr + " & " + alg],
label=alg)
     ax.set(xlabel='Sample sizes', ylabel='Nanoseconds',
           title='Time measures of ' + curr)
     plt.legend()
fig.savefig("images/OneAddComputingTime.png")
plt.show()
                                                      1e6 Time measures of Binary Large Text
              Time measures of Binary Small Text
     300000
                Naive Comparison
                                                           Naive Comparison
                Standard KMP
                                                           Standard KMP
                                                   2.6
    280000
                Optimal KMP
                                                           Optimal KMP
    260000
                                                   2.4
    240000
   Nanoseconds
                                                 Nanoseconds
                                                   2.2
    220000
                                                   2.0
    200000
    180000
                                                   1.8
    160000
                                                   1.6
    140000
                                                                     1500 2000
                500
                     1000
                          1500
                                2000
                                     2500
                                           3000
                                                           500
                                                                1000
                                                                                 2500
                                                                                      3000
                        Sample sizes
                                                                   Sample sizes
                                                      <sub>1e6</sub> Time measures of DNA Large Text
               Time measures of DNA Small Text
                Naive Comparison
                                                           Naive Comparison
                Standard KMP
                                                           Standard KMP
     300000
                Optimal KMP
                                                           Optimal KMP
                                                   1.6
    250000
                                                Nanoseconds
1.4
   Nanoseconds
    200000
                                                   1.3
    150000
                                                   1.2
    100000
                                                   1.1
                          1500
                                2000
                                      2500
                                                                     1500
                                                                           2000
                500
                     1000
                                           3000
                                                           500
                                                                1000
                                                                                 2500
                        Sample sizes
                                                                   Sample sizes
```

#@title Графики количества посимвольных сравнений по всем алгоритмам для конкретного текста, где в шаблоне заменен 1 символ на '?' #for all algs, operations table with 1 add of '?' in samples

```
gs = gridspec.GridSpec(2, 2)
fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(15, 10))
for curr, grd in zip(texts, itertools.product([0, 1], [0, 1])):
    ax = plt.subplot(gs[grd[0], grd[1]])
    for alg in algs:
        plt.plot(oper["Sample sizes"], oper[curr + " & " + alg],
label=alg)
    ax.set(xlabel='Sample sizes', ylabel='Quantity',
        title='Symbol comparison of ' + curr)
    plt.legend()

fig.savefig("images/OneAddOperations.png")
plt.show()
```



Графики для шаблонов с 2 измененными символами

```
times = pd.read_csv("tables/БПИ213_KaдыкoBa_2AddsComputingTime.csv", sep=';')
oper = pd.read_csv("tables/БПИ213_KaдыкoBa_2AddsOperations.csv", sep=';')
#@title Графики измеренного времени по всем алгоритмам для конкретного текста, где в шаблоне заменены 2 символа на '?'
#for all algs, computing time table with 2 add of '?' in samples

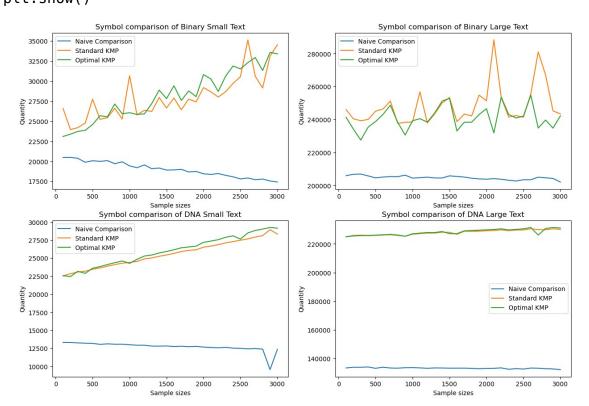
qs = qridspec.GridSpec(2, 2)
```

```
fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(10, 10))
for curr, grd in zip(texts, itertools.product([0, 1], [0, 1])):
     ax = plt.subplot(gs[grd[0], grd[1]])
     for alg in algs:
           plt.plot(times["Sample sizes"], times[curr + " & " + alg],
label=alg)
     ax.set(xlabel='Sample sizes', ylabel='Nanoseconds',
           title='Time measures of ' + curr)
     plt.legend()
fig.savefig("images/TwoAddsComputingTime.png")
plt.show()
                                                      1e6 Time measures of Binary Large Text
              Time measures of Binary Small Text
     400000
                Naive Comparison
                                                                            Naive Comparison
                Standard KMP
                                                                            Standard KMP
                                                   2.8
                Optimal KMP
                                                                            Optimal KMP
     350000
                                                   2.6
                                                 Nanoseconds
2.2
2.0
    300000
   Nanoseconds
    250000
    200000
                                                   1.8
                                                   1.6
    150000
                          1500
                                2000
                                      2500
                                           3000
                                                                      1500
                                                                           2000
           0
                500
                     1000
                                                           500
                                                                1000
                                                                                 2500
                                                                                       3000
                        Sample sizes
                                                                    Sample sizes
               Time measures of DNA Small Text
                                                      1e6 Time measures of DNA Large Text
                Naive Comparison
                                                                            Naive Comparison
    260000
                                                   1.8
                                                                            Standard KMP
                Standard KMP
                Optimal KMP
                                                                            Optimal KMP
    240000
                                                   1.7
    220000
                                                   1.6
                                                 Nanoseconds
   Nanoseconds
    200000
    180000
                                                   1.4
    160000
    140000
                                                   1.3
    120000
                                                   1.2
    100000
                                2000
                     1000
                          1500
                                      2500
                                           3000
                                                                      1500
                                                                          2000
                                                                                 2500
                                                                                       3000
                500
                                                      0
                                                           500
                                                                1000
                        Sample sizes
                                                                    Sample sizes
```

#@title Графики количества посимвольных сравнений по всем алгоритмам для конкретного текста, где в шаблоне заменены 2 символа на '?' #for all algs, operations table with 2 adds of '?' in samples

```
fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(15, 10))
for curr, grd in zip(texts, itertools.product([0, 1], [0, 1])):
    ax = plt.subplot(gs[grd[0], grd[1]])
    for alg in algs:
        plt.plot(oper["Sample sizes"], oper[curr + " & " + alg],
label=alg)
    ax.set(xlabel='Sample sizes', ylabel='Quantity',
        title='Symbol comparison of ' + curr)
    plt.legend()

fig.savefig("images/TwoAddsOperations.png")
plt.show()
```



Графики для шаблонов с 3 измененным символом

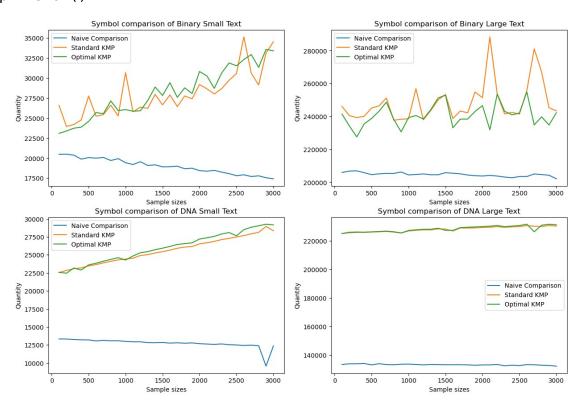
```
for alg in algs:
            plt.plot(times["Sample sizes"], times[curr + " & " + alg],
label=alg)
      ax.set(xlabel='Sample sizes', ylabel='Nanoseconds',
            title='Time measures of ' + curr)
      plt.legend()
fig.savefig("images/ThreeAddsComputingTime.png")
plt.show()
                                                           <sub>1e6</sub>Time measures of Binary Large Text
               Time measures of Binary Small Text
     300000
                  Naive Comparison
                                                                                   Naive Comparison
                                                       2.4
                 Standard KMP
                                                                                   Standard KMP
     280000
                                                                                   Optimal KMP
                 Optimal KMP
     260000
                                                       2.2
     240000
   Nanoseconds
                                                     Nanoseconds
     220000
                                                       2.0
     200000
                                                       1.8
     180000
     160000
                                                        1.6
     140000
                 500
                       1000
                             1500
                                   2000
                                         2500
                                               3000
                                                                500
                                                                      1000
                                                                            1500
                                                                                  2000
                                                                                        2500
                                                                                              3000
                          Sample sizes
                                                                          Sample sizes
                                                           <sub>1e6</sub> Time measures of DNA Large Text
                Time measures of DNA Small Text
                                    Naive Comparison
                                                       1.9
                                                                                   Naive Comparison
     250000
                                    Standard KMP
                                                                                   Standard KMP
                                    Optimal KMP
                                                                                   Optimal KMP
                                                       1.8
     225000
                                                       1.7
     200000
   Nanoseconds
                                                     Nanoseconds
                                                       1.6
     175000
                                                        1.5
     150000
     125000
                                                       1.3
     100000
                                                        1.2
                                                        1.1
                 500
                       1000
                             1500
                                   2000
                                         2500
                                               3000
                                                                500
                                                                      1000
                                                                            1500
                                                                                  2000
                                                                                        2500
```

#@title Графики количества посимвольных сравнений по всем алгоритмам для конкретного текста, где в шаблоне заменены 3 символа на '?' #for all algs, operations table with 3 adds of '?' in samples

Sample sizes

```
gs = gridspec.GridSpec(2, 2)
fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(15, 10))
for curr, grd in zip(texts, itertools.product([0, 1], [0, 1])):
    ax = plt.subplot(gs[grd[0], grd[1]])
```

Sample sizes



Графики для шаблонов с 4 измененным символом

```
times = pd.read_csv("tables/БПИ213_Kадыкова_4AddsComputingTime.csv", sep=';')

oper = pd.read_csv("tables/БПИ213_Kадыкова_4AddsOperations.csv", sep=';')

#@title Графики измеренного времени по всем алгоритмам для конкретного текста, где в шаблоне заменены 4 символа на '?'

#for all algs, computing time table with 4 add of '?' in samples

gs = gridspec.GridSpec(2, 2)

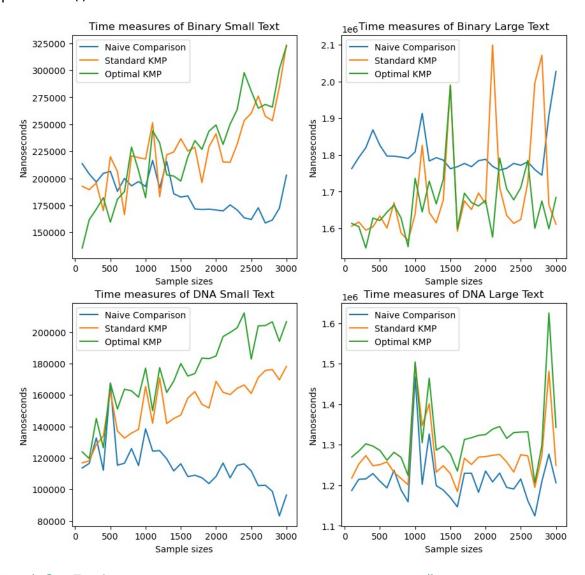
fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(10, 10))

for curr, grd in zip(texts, itertools.product([0, 1], [0, 1])):

    ax = plt.subplot(gs[grd[0], grd[1]])
    for alg in algs:
        plt.plot(times["Sample sizes"], times[curr + " & " + alg], label=alg)
```

```
ax.set(xlabel='Sample sizes', ylabel='Nanoseconds',
    title='Time measures of ' + curr)
plt.legend()
```

fig.savefig("images/FourAddsComputingTime.png")
plt.show()



#@title Графики количества посимвольных сравнений по всем алгоритмам для конкретного текста, где в шаблоне заменены 4 символа на '?' #for all algs, operations table with 4 adds of '?' in samples

```
gs = gridspec.GridSpec(2, 2)
fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(15, 10))
for curr, grd in zip(texts, itertools.product([0, 1], [0, 1])):
    ax = plt.subplot(gs[grd[0], grd[1]])
    for alg in algs:
        plt.plot(oper["Sample sizes"], oper[curr + " & " + alg],
label=alg)
```

```
ax.set(xlabel='Sample sizes', ylabel='Quantity',
    title='Symbol comparison of ' + curr)
plt.legend()
```

fig.savefig("images/FourAddsOperations.png")
plt.show()

