# Kazakh-British Technical University

# Algorithms and Data Structures, Spring 2011

## Lecture 8: Text Processing

Read the following chapters: [chapter 32: 32.1, 32.2, 32.4] Thomas H. Cormen, Charles E. Leiserson. *Introduction to algorithms – 2-nd edition.* – USA : MIT Press, 2001. – 1180p.

## 1 How to read text?

Example 1. Option 1.

```
#include <iostream>

using namespace std;

int main(){
  freopen("input.txt", "r", stdin);//read from input file
  freopen("output.txt", "w", stdout);//write to output file
  string s;
  while (cin >> s)
    cout << s;
  return 0;
}
```

input.txt

```
Hello world!
Algorithms and Data Structures.
```

output.txt

```
Helloworld!AlgorithmsandDataStructures.
```

Example 2. Option 2.

```
#include <iostream>

using namespace std;

int main(){
  freopen("input.txt", "r", stdin);//read from input file
  freopen("output.txt", "w", stdout);//write to output file
  string s;
  while (getline(cin,s))
    cout << s << endl;
  return 0;
}
```

input.txt

```
Hello world!
Algorithms and Data Structures.
```

output.txt

```
Hello world!
Algorithms and Data Structures.
```

## 2 What are the problems in text processing?

Modification of text
Add new information
Remove some part of the text
String matching

## 3 String operations

"abc" + "def" = "abcdef"

Prefix (beginning): "abcdef"
Suffix (ending): "abcdef"

## 4 String matching

Problem: Given string $S$ and text $T$. Task is to find all occurrences of $S$ in $T$.

Existing algorithms:
Naive (brute-force) algorithm $O(mn)$
Rabin-Karp algorithm $O(m+n)$
Knuth-Morris-Pratt algorithm $O(m+n)$

## References

[1] [chapter 32] Thomas H. Cormen, Charles E. Leiserson. *Introduction to algorithms – 2-nd edition.* – USA : MIT Press, 2001. – 1180p.