

ШИФРУВАННЯ ДАНИХ НА МІКРОПРОЦЕСОРАХ ЗА ДОПОМОГОЮ МОВИ ПРОГРАМУВАННЯ JAVA

Об'єкт дослідження, предмет та мета

- ◎ Об'єктом дослідження є найпоширеніші алгоритми шифрування і мова програмування Java.
- ◎ Предметом дослідження є існуючі криптографічні системи, можливості об'єктно орієнтованої мови програмування Java та новий метод оцінки криптографічних систем.
- ◎ Мета роботи полягає у розробці нового методу шифрування.

Проблеми, що пов'язані з захистом інформації

- ◎ В наш час існують дві найбільші проблеми, що пов'язані з захистом інформації. По-перше, немає дійсно якісного алгоритму шифрування з гарним співвідношенням ціни-якості.
- ◎ По-друге, до сих пір не розроблені методики оцінки ефективності криптографічних систем, враховуючи ціну-якість.

Методика оцінки ефективності криптографічної системи

- ◎ Простота використання
- ◎ Швидкість шифрування інформації
- ◎ Стійкість алгоритму до зовнішніх атак
- ◎ Ціна
- ◎ Кількість використаних математичних операцій
- ◎ Перспективність алгоритму

Новий метод шифрування

Алгоритм складається з двох частин:

- ⦿ шифрування даних;
- ⦿ розшифрування інформації за допомогою секретного ключа.

Суть методу шифрування

Винайдений новий метод шифрування на мікропроцесорах було виконано на об'єктно орієнтованій мові програмування Java. Суть цієї криптографічної системи полягає в тому, що кожний наступний елемент, який потрібно зашифрувати, шифрується попереднім.

Для прикладу взято повідомлення, яке складається з 5 букв «р» «а» «s» «h» «а».

Спочатку, шифрування починається з кодування символів (в нашому випадку англійського алфавіту), тут використано стандартне кодування в Windows Аски (ASCII). Від А до Z – від 97 до 122.

Шифрування на Java виконується наступним чином. Спочатку кожній букві англійської абетки привласнюють цифри від 97 до 122. Далі пишеться код для виведення повідомлення «Введіть ключ» і створюється поле для вводу інформації, а саме для вводу ключа в вигляді цифри. Схожий код написаний і для вводу повідомлення, після чого виконується цикл, який виводить на екран зашифроване повідомлення по формулі: код букви повідомлення + ключ + попередній код букви (якщо така є). В результаті отримуємо зашифроване повідомлення у вигляді цифр.

Дешифрування в мові програмування Java виконується таким чином: спочатку написаний код виводить повідомлення на екран з проханням ввести ключ і відкривається поле для його введення. Схожий код написаний і для введення зашифрованого повідомлення, яке потрібно розшифрувати. Після цього виконується цикл, який по формулі: код букви повідомлення мінус код попередньої букви повідомлення (якщо така є) мінус ключ (для першого символу повідомлення) дешифрує повідомлення. В результаті отримуємо розшифрований текст.

Частини коду

```
/* Присвоение числа каждой букве алфавита: a=97,b=98,c=99,d=100,e=101,f=102,g=103,h=104,i=105,j=106,k=107,l=108,  
m=109,n=110,o=111,p=112,q=113,r=114,s=115,t=116,u=117,v=118,w=119,x=120, y=121,z=122*/
```

```
public static void main(String arg[]){  
    // Вводим ключ  
    System.out.println("Введите ключ");  
    Scanner pervoe = new Scanner(System.in);  
    int kluch = pervoe.nextInt();  
    // Вводим первую букву  
    System.out.println("Введите букву 1");  
    Scanner scan1 = new Scanner(System.in);  
    char bykva1 = scan1.next().charAt(0);  
    int perviyvvod = bykva1+kluch;  
  
    // Цикл для шифрования первой буквы  
    if (bykva1=='a'){switch (bykva1){  
        case 'a': bykva1=1; break;} System.out.println(perviyvvod);}  
    else if (bykva1=='b'){switch (bykva1){  
        case 'b': bykva1=2; break;} System.out.println(perviyvvod);}  
  
    // Цикл для шифрования второй буквы  
    System.out.println("Введите букву 2");  
    Scanner scan2 = new Scanner(System.in);  
    char bykva2 = scan2.next().charAt(0);  
    int vtoroyvvod = perviyvvod + bykva2;  
  
    switch (bykva2) {  
        case 'a': System.out.println(vtoroyvvod); break;  
        case 'b': System.out.println(vtoroyvvod); break;  
  
        System.out.println("Ваш код: " + perviyvvod + "," + vtoroyvvod + "," + tretiyvvod + "," + chetvertiyvvod + "," + pjatoyvvod);} }
```


Результат работы алгоритму шифрования

The screenshot displays the IntelliJ IDEA IDE interface. The main editor shows the source code for a Java class named `Shifrovanie`. The code includes an import for `java.util.*` and a comment in Russian: `/*Присвоение числа каждой букве алфавита a=97,b=98,c=99,d=100,e=101,f=102,g=103,h=104,i=105,j=106,k=107,l=108,m=109,n=110,o=111,p=112,q=113,r=114,s=115`. The `main()` method is highlighted.

The Run window at the bottom shows the execution output for the `Shifrovanie` class. The output consists of prompts for input and the resulting code values:

```
"C:\Program Files\Java\jdk1.8.0_102\bin\java" ...
Введите ключ
2
Введите букву 1
P
114
Введите букву 2
A
211
Введите букву 3
X
326
Введите букву 4
B
430
Введите букву 5
A
527
Ваш код: 114,211,326,430,527
Process finished with exit code 0
```

Результат работы алгоритму дешифрования

The screenshot displays the IntelliJ IDEA IDE interface. The top toolbar includes menus for File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, and Help. The project structure on the left shows a project named 'Diplom' with subdirectories 'idea', 'out', and 'src'. The 'src' directory contains 'Deshifrovanie' and 'Shifrovanie' packages. The main editor window shows the 'Deshifrovanie.java' file with the following code:

```
1 import java.util.*;
2 public class Deshifrovanie extends Shifrovanie {
3
4
5     public static void main(String arg[]) {
6
7         // ВВОДИМ КЛЮЧ
8         System.out.println("Введите ключ");
```

The 'Run' console at the bottom shows the execution output:

```
"C:\Program Files\Java\jdk1.8.0_102\bin\java" ...
Введите ключ
2
Введите цифру 1
114
112
Введите цифру 2
211
99
Введите цифру 3
326
115
Введите цифру 4
430
104
Введите цифру 5
527
97
Зашифрованное сообщение:
p
a
s
h
a
Process finished with exit code 0
```

Висновки

В випускній роботі було розглянуто нову методику оцінки ефективності криптографічних систем, яка дає змогу замовнику обрати найкращий алгоритм шифрування, враховуючи його потреби. Простота, корисність, а головне – точність даної методики, робить необхідним її використання для вибору алгоритму шифрування.

Також, було розроблено новий метод шифрування інформації, який пройшов оцінку ефективності криптографічних систем, отримав оцінку 7 і рекомендований до застосування на підприємстві.

ДЯКУЮ ЗА УВАГУ!