Модулярные шифры

Кевролетин В.В.

12 декабря 2011 г.

Задание

Условие

Расшифровать заданное сообщение ymjkw jvzjs hdrjy mtisj jixqt slhnu mjwyj cyxyt btwp с использованием частотной таблицы (модулярный шифр с n=1).

Решение

k = 21 открытый текст: the frequency method needs long cipher texts to work

Задание

Условие

Показать, что нод(m,n)=1 н. и д. для однозначности дешифров ания модулярного шифра.

Решение

Необходимость:

От противного: для заданных n = 10, m = 26 посчитаем значение шифра для чисел 0 и 13: НОД(10, 26) = 2
0*10(mod 26) = 0
13*10(mod 26) = 130(mod 26) = 0
Получили, противоречие.

Достаточность:
От противного:
Пусть есть 2 числа a,b такие что 0 <= a < 26, 0 < =b < 26, a не равно b и a*n(mod m) = b*n(mod m)
(a - b)*n(mod m) = 0(mod m)
Но т.к. НОД(n,m) = 1, то НОК(n,m) = n*m следовательно
(a - b)*n >= n*m
т.е. (a - b) > m - Противоречие, т.к. a < m и b Б m по условию.

Задание

Условие

Описать обратное преобразование для модулярного шифра с n!=1. Будет ли оно модулярным шифром.

Решение

Рассмотрис сначала случай k = 0: Как было показано выше из НОД(n,m) следует однозначность шифрования. Это значит что каждому элементу от 0 до m-1 будет сопоставлено единственное число в диапазоне 0 .. m-1. Тогда айдется такой x, что

$$1 = x * n(mod m)$$

т.е. существует элемент обратный к n: $n^{-1} = x$ Тогда если мы хоти расшифровать число b, полученное следующим образом:

$$b = a * n(mod m)$$

Т.е. хотим узнать число a. Достаточно умножить обе части на n^{-1}:

$$b * n^{-1} = a * n * n^{-1}(mod\ m) b * n^{-1} = a(mod\ m)$$

По сути процесс дешифрования аналогичеу шифрованию и является модулярным шифром.

При к != 0:

$$1 = x * n + k(mod\ m)$$

$$1 - k = x * n(mod\ m)$$

$$b = a * n + k(mod\ m)$$

$$b - k = a * n(mod\ m)$$

$$b - k = a * (1 - k)(mod\ m)$$

$$(b - k) * (1 - k)^{-1} = a(mod\ m)$$

Задание

Условие

Описать методику криптоанализа модулярного шифра с n!=1.

Решение

Сначала необходимо подсчитать частоты появления каждого символа в криптотексте. Распределение букв в криптотексте затем надо сравнить с распределением букв в алфавите исходных сообщений. Буква с наивысшей частотой в криптотексте соответствует букве наивысшей частотой в алфавите исходного сообщения, и т.д. для менее часто встречающихся символов.

Задание

Условие

Реализовать программу (Mathematica, Scheme, Sage, ...) для подсчета частоты встречаемости отдельных символов, пар, троек и т.д. Подготовить тесты. Продемонстрировать работу на достаточно длинном тексте. Сравнить результаты с известными.

Решение

```
sub split_to_groups {
    my ($text, $len) = @_;
    $len ||= 1;
    my @a = split //, $text;
    my @res;
    my $i = 0;
    while ($i < @a) {
        my @t;
        for (1 .. $len) {
            push @t, $a[$i++];
            last if $i == @a
        }
        push @res, join '', @t;
    }
    \@res
}

sub occurrence_freq {
    my ($text, $len) = @_;
    my %h;
    for (@{split_to_groups($text, $len)}) {
        $h{$_} = 0 unless defined $h{$_};
        ++$h{$_};
    }
    \%h
}

my $text = join '', qw(
QN FS LK CM LT HC SM MC VK
IH HA XR QM BQ IE QN AK RD
PS TU CB NX MC IF NX MC IT
YF SD EF IF QN LQ FL YD SB
QN AK EU MC TI IE QN MS IQ
KA PF IL BM WD DF RE IV KA
MC IT QN FX MB FT FT DX AK
HC SM YF WE BA AB QE IV OI
XT IT FM AQ AK QN MX ZU DS
OI XI QN FY RX NV OR RB RA
MC MB NX XM AE OW FT LR NC
```

```
IQ QN FM ML SN AH QN QL TW
FL ST LT PI QI QN DS VK AR
FS AQ TI DF SM AK FO XM VA
RZ FT SN GS UD FM SA WA LN
MF IT QN FG LN BQ QE AR VA
DT FT QA AB FY IT MX DK FM
DF QN FX NO XC TF SM FK OY
CM QM BA LH
);

sub statistics {
    my ($text, $len, $min_cnt) = @_;
    my $res = occurrence_freq($text, $len);
    my $sum = sum values %$res;
    my @pairs = map { [$_, $res->{$_}, $res->{$_} * 100 / $sum] }
        grep { $res->{$_} >= $min_cnt } keys %$res;
    for (sort { $b->[1] <=> $a->[1] } @pairs) {
        printf "%s: %5.d   - %.2f %%\n", @{$_}
    }
}
```

Проверено на примере из книги Саломаа А. "КРИПТОГРАФИЯ С ОТКРЫТЫМ КЛЮЧОМ"стр. 40 с разбиением текста на пары символов. Результат выполнения функции (криптотекс, хранящийся в пременной $text упущен,его можно посмотреть в книге и моей 1й работе, выведена статистика для пар, входящих в криптотекст более 3х раз) соответствует данным в книге

$$statistics(\$text, 2, 4)$$

```
QN:    13   - 7.83 %
MC:     6   - 3.61 %
IT:     5   - 3.01 %
FT:     5   - 3.01 %
AK:     5   - 3.01 %
SM:     4   - 2.41 %
FM:     4   - 2.41 %
```

А так же для примера со страницы 34, где требовалось посчитать статистику для каждого символа.

```
U:    32   - 13.28 %
C:    31   - 12.86 %
Q:    23   - 9.54 %
F:    21   - 8.71 %
V:    20   - 8.30 %
P:    15   - 6.22 %
I:    15   - 6.22 %
T:    14   - 5.81 %
A:     8   - 3.32 %
X:     8   - 3.32 %
N:     7   - 2.90 %
```

K: 7 – 2.90 %
E: 7 – 2.90 %
M: 6 – 2.49 %
R: 6 – 2.49 %
Z: 5 – 2.07 %
B: 5 – 2.07 %
D: 4 – 1.66 %
W: 3 – 1.24 %
Y: 2 – 0.83 %
H: 1 – 0.41 %
G: 1 – 0.41 %

Это так же соответствует результатам из книги.

Задание

Условие

Сколько всего различных модулярных шифров в m-буквенном алфавите (в английском языке, m=26)?

Решение

Шинфр a -> b b = a*n + k Шифр полностью определятеся парой чисел n и k, Существует 12 простых чисел в промежутке от 1 до 26: 1, 3, 5, 7, 9, 11, 15, 17, 19 21, 23, 25. Существует 26 возможных значений для k, причем они мо- гут быть выбраны независимо от значений для a, за исключением случая a = 1, b = 0. Это дает в совокупности $12 \cdot 26 - 1 = 311$ шифров.