

## АНАЛИЗ НА РЕШЕНИЕТО НА ЗАДАЧА D2. ДЕКОДИРАНЕ

Прочитаме кодираното съобщение от входа. Отделяме цифровите знаци в началото и всяка цифра подменяме с допълнението ѝ до 9. След това сглобяваме числото  $d$  от съставните му декодирани цифри. Същото правим и с цифровите символи в края на кодираното съобщение, от които сглобяваме числото  $s$ . Като премахнем цифровите символи от двата края на кодираното съобщение, неговата дължина остава  $n$  – колкото е дължината на оригиналното съобщение. Разделяме кодираното съобщение на части, които съответстват на отделните редове получени в процеса на кодиране. Ако  $n$  ератно на  $d$ , всички редове с текст при кодирането съдържат еднакъв брой символи ( $n \text{ div } d$ ). Ако  $n$  не ератно на  $d$ , няколко реда при кодирането имат по един символ в повече. Броят на тези редове е остатъкът при целочисленото делене на  $n$  и  $d$  и са първите по номер редове при кодирането. Последния знак в ред  $r$  при кодирането е  $r$ -ия знак от некодирания текст, предходения му знак в реда с кодиран текст е с номер  $r+d$  в некодирания текст. Използваме тази зависимост, за да определим всеки пореден знак в участък от кодираното съобщение, какъв номер е имал в оригиналното съобщение. Прехвърляме всеки знак от кодираното съобщение на съответното му място в разкодираното съобщение. Накрая печатим разкодираното съобщение.

### Сорс-кодът на програмата

```
#include <iostream>
#include <cstring>
#include <cctype>
using namespace std;
#define MAXLEN 10000

char A[MAXLEN+1], /// Низът с декодираното (оригиналното) съобщение
      B[MAXLEN+20]; /// Низът с кодираното съобщение
int d,          /// брой символи по обиколката на пръчката за кодиране
    n,          /// дължина на оригиналното съобщение
    s,          /// номер на реда с текст, който е в началото на кодирания текст
    cr,         /// номер на ред с текст, който текущо се декодира
    len,        /// стандартна дължина на ред (участък) за декодиране (len=n/d)
    fr,         /// брой кодирани редове с по-голяма с 1 дължина (fr=n%d)
    j,          /// място в декодираното съобщение, където да се копира
                /// поредния символ от кодираното съобщение
    i;          /// номер на пореден обработваем символ от кодираното съобщение

int main() {

    cin.getline(B,MAXLEN+20);

    /// Кой е стартовият ред при кодирането
    n = strlen(B);
    for (i=n-1, j=1; isdigit(B[i]); j*=10, i--) s += j*('9'-B[i]);
    s--;          /// Преобразуваме в стил на броене „от нула“
    n = i+1;     /// Елиминираме цифровите символи в края на кодираното съобщение

    /// Колко е d
    for (i=0; isdigit(B[i]); i++) d = d*10 + ('9'-B[i]);
    n-=i;        /// Елиминираме цифровите символи в началото на кодираното съобщение
    len = n/d;
    fr = n%d;
    cr = s;
```

```
do {  
    for (j=cr+d*(len-1+(fr>cr)); j>=0; j-=d) A[j] = B[i++];  
    cr = ++cr%d;  
} while (cr != s);  
  
cout << A << endl;  
return 0;  
}
```

*Автор: Евгений Василев*