

REMÉNYEINK ÉS FÉLELMEINK

*Probojcsevity Ottó, IT szakértő
2020. november 6.*



A digitális felzárkózás új szintre emeli a települési vízgazdálkodást!

REMÉNYEK

- *Sok-sok új technológia*
 - *Olcsó, könnyen telepíthető*
 - *Sokkal jobb érzékekkel és cselekvési sebességgel, mint eddig bármi*
 - *Helyettünk is gyors, szorgos és okos*
 - *Bűvszavak: IoT, LoRa, 5G, AI, Big Data (Machine Learning!... „csirkekehús”)*
- *Sok-sok új válasz*
 - *Elosztott rendszerek*
 - *Az érzékek és vezérlések közös felülete*
 - *Növekvő megbízhatóság*



ESZKÖZÖK...

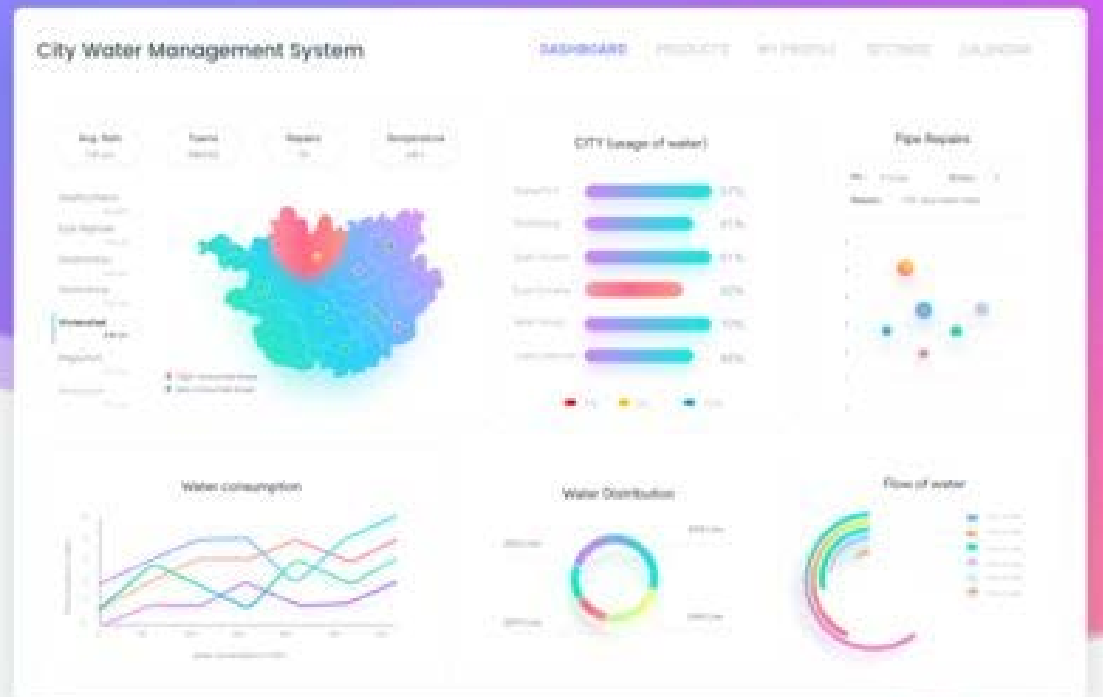


REMÉNYEK

- *Sok-sok új technológia*
 - *Olcsó, könnyen telepíthető*
 - *Sokkal jobb érzékekkel és cselekvési sebességgel, mint eddig bármi*
 - *Helyettünk is gyors, szorgos és okos*
 - *Bűvszavak: IoT, LoRa, 5G, AI, Big Data (Machine Learning!... „csirkekehús”)*
- *Sok-sok új válasz*
 - *Elosztott rendszerek*
 - *Az érzékek és vezérlések közös felülete*
 - *Növekvő megbízhatóság*



FELÜLETEK...



REMÉNYEK

- *Sok-sok új technológia*
 - *Olcsó, könnyen telepíthető*
 - *Sokkal jobb érzékekkel és cselekvési sebességgel, mint eddig bármi*
 - *Helyettünk is gyors, szorgos és okos*
 - *Bűvszavak: IoT, LoRa, 5G, AI, Big Data (Machine Learning!... „csirkekehús”)*
- *Sok-sok új válasz*
 - *Elosztott rendszerek*
 - *Az érzékek és vezérlések közös felülete*
 - *Növekvő megbízhatóság*



FÉLELMEINK

- *Rengeteg a hiányzó tudás*
 - *Hogyan kell megcsinálni?*
 - *Ki csinálja meg?*
 - *Miből?*
 - *Tényleg olyan jó? (pl. biztonság)*
 - *Ismeretlen terület – ismeretlen kockázat*
 - *Mi az, ami időtálló, és mi az ami már túlhaladott? (pl. TETRA)*
- *Új kihívások*
 - *Túlnépesedés*
 - *Globális felmelegedés*
 - *Megváltozott társadalmi környezet*



AKKOR MOST MERRE?...

(... JAVASLATOK, AMIK TALÁN ITT IS MŰKÖDNEK)

- *Adaptáció*
 - *A burjánzó komplementer eszközpiac követése*
 - *Ami megvan és „olcsó”, az azonnal terjedjen el - ez tud forrásokat teremteni az erősen kontrollált iparágban*
 - *Tág szakmai értékelvűség, döntéshozatal*
 - *Az új társadalmi szabályok új megoldásai*
 - *Vertikális megosztás -> óvatosan! (római ló*)*
- *Tudás építés*
 - *Képzések*
 - *Pilotok*
 - *IT bevonása, beépítése (Audi TT műszerfal)*
 - *Nagy megbízhatóságú rendszerek tervezése (Ős-elmélet: „Reed–Solomon”)*



AKKOR MOST MERRE?...

(... JAVASLATOK, AMIK TALÁN ITT IS MŰKÖDNEK)

- Adaptáció



„Kecé Solomon”



éle

AKKOR MOST MERRE?...

(... JAVASLATOK, AMIK TALÁN ITT IS MŰKÖDNEK)

- *Adaptáció*
 - *A burjánzó komplementer eszközpiac követése*
 - *Ami megvan és „olcsó”, az azonnal terjedjen el - ez tud forrásokat teremteni az erősen kontrollált iparágban*
 - *Tág szakmai értékelvűség, döntéshozatal*
 - *Az új társadalmi szabályok új megoldásai*
 - *Vertikális megosztás -> óvatosan! (római ló*)*
- *Tudás építés*
 - *Képzések*
 - *Pilotok*
 - *IT bevonása, beépítése (Audi TT műszerfal)*
 - *Nagy megbízhatóságú rendszerek tervezése (Ős-elmélet: „Reed–Solomon”)*



ÚJ TÁRSADALOM?...



AKKOR MOST MERRE?...

(... JAVASLATOK, AMIK TALÁN ITT IS MŰKÖDNEK)

- *Adaptáció*
 - *A burjánzó komplementer eszközpiac követése*
 - *Ami megvan és „olcsó”, az azonnal terjedjen el - ez tud forrásokat teremteni az erősen kontrollált iparágban*
 - *Tág szakmai értékelvűség, döntéshozatal*
 - *Az új társadalmi szabályok új megoldásai*
 - *Vertikális megosztás -> óvatosan! (római ló*)*
- *Tudás építés*
 - *Képzések*
 - *Pilotok*
 - *IT bevonása, beépítése (Audi TT műszerfal)*
 - *Nagy megbízhatóságú rendszerek tervezése (Ős-elmélet: „Reed–Solomon”)*





The Home of Remanufacturing



AKKOR MOST MERRE?...

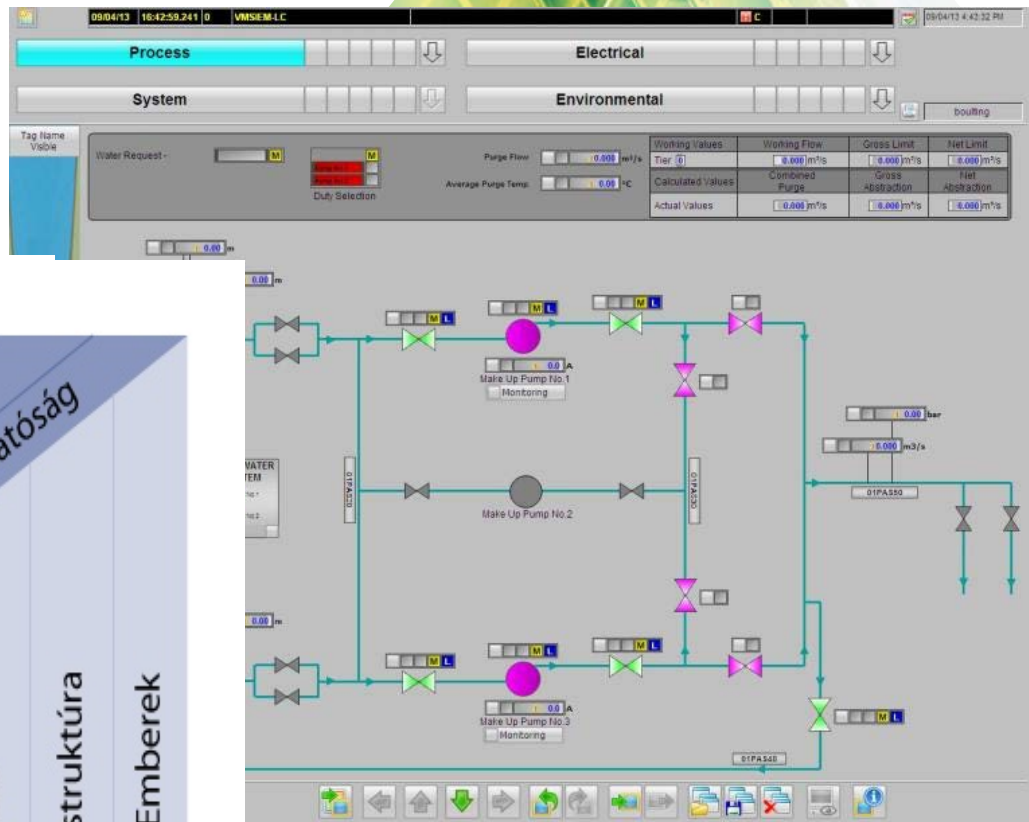
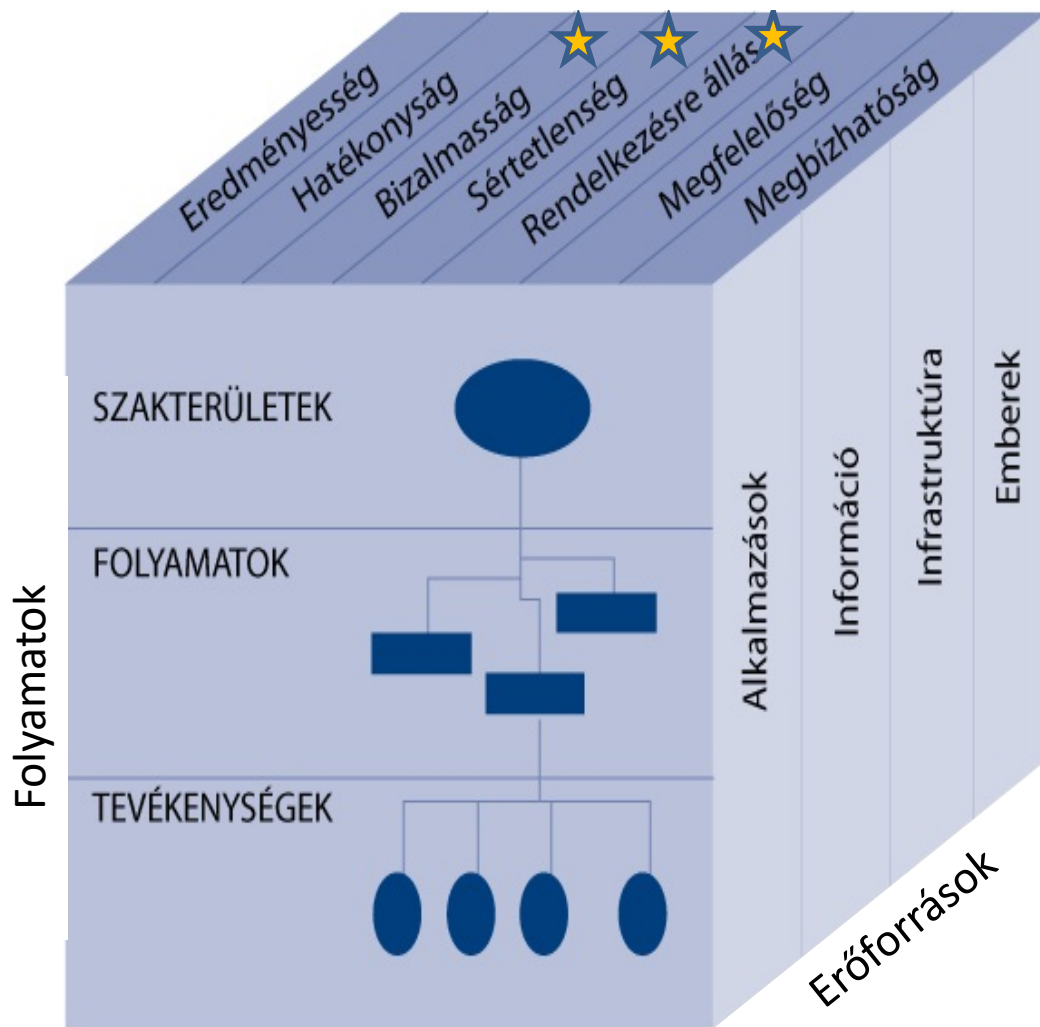
(... JAVASLATOK, AMIK TALÁN ITT IS MŰKÖDNEK)

- *Adaptáció*
 - *A burjánzó komplementer eszközpiac követése*
 - *Ami megvan és „olcsó”, az azonnal terjedjen el - ez tud forrásokat teremteni az erősen kontrollált iparágban*
 - *Tág szakmai értékelvűség, döntéshozatal*
 - *Az új társadalmi szabályok új megoldásai*
 - *Vertikális megosztás -> óvatosan! (római ló*)*
- *Tudás építés*
 - *Képzések*
 - *Pilotok*
 - *IT bevonása, beépítése (Audi TT műszerfal)*
 - *Nagy megbízhatóságú rendszerek tervezése (Ős-elmélet: „Reed–Solomon”)*



BIZTONSÁG?...

Követelmények



REED-SOLOMON?...



W Reed-Solomon error correction - Wikipedia

https://en.wikipedia.org/wiki/Reed-Solomon_error_correction#:~:text=Reed-Solomon%20codes%20are%20a,and%20Gustave%20Solomon%20in%20196...

Lili JHC Koronavírus-megbe... A Swiss Doctor on... ATL EURO MOMO COVID-19 | Analytics Covid Trends Downdetector Medium - Get smar... MEGA Free 3D Modeling S... SketchUp Telenor (Hungary) ... Other favorites

Reed & Solomon's original view: The codeword as a sequence of values [\[edit\]](#)

There are different encoding procedures for the Reed–Solomon code, and thus, there are different ways to describe the set of all codewords. In the original view of [Reed & Solomon \(1960\)](#), every codeword of the Reed–Solomon code is a sequence of function values of a polynomial of degree less than k . In order to obtain a codeword of the Reed–Solomon code, the message is interpreted as the description of a polynomial p of degree less than k over the finite field F with q elements. In turn, the polynomial p is evaluated at $n \leq q$ distinct points $a_1, \dots, a_n$ of the field F , and the sequence of values is the corresponding codeword. Common choices for a set of evaluation points include $\{0, 1, 2, \dots, n-1\}$, $\{0, 1, \alpha, \alpha^2, \dots, \alpha^{n-2}\}$, or for $n < q$, $\{1, \alpha, \alpha^2, \dots, \alpha^{n-1}\}$, ..., where α is a [primitive element](#) of F .

Formally, the set $\mathbf{C}$ of codewords of the Reed–Solomon code is defined as follows:

$$\mathbf{C} = \left\{ (p(a_1), p(a_2), \dots, p(a_n)) \mid p \text{ is a polynomial over } F \text{ of degree } < k \right\}.$$

Since any two *distinct* polynomials of degree less than k agree in at most $k-1$ points, this means that any two codewords of the Reed–Solomon code disagree in at least $n - (k-1) = n - k + 1$ positions. Furthermore, there are two polynomials that do agree in $k-1$ points but are not equal, and thus, the [distance](#) of the Reed–Solomon code is exactly $d = n - k + 1$. Then the relative distance is $\delta = d/n = 1 - k/n + 1/n = 1 - R + 1/n \sim 1 - R$, where $R = k/n$ is the rate. This trade-off between the relative distance and the rate is asymptotically optimal since, by the [Singleton bound](#), every code satisfies $\delta + R \leq 1 + 1/n$. Being a code that achieves this optimal trade-off, the Reed–Solomon code belongs to the class of [maximum distance separable codes](#).

While the number of different polynomials of degree less than k and the number of different messages are both equal to q^k , and thus every message can be uniquely mapped to such a polynomial, there are different ways of doing this encoding. The original construction of [Reed & Solomon \(1960\)](#) interprets the message x as the *coefficients* of the polynomial p , whereas subsequent constructions interpret the message as the *values* of the polynomial at the first k points $a_1, \dots, a_k$ and obtain the polynomial p by interpolating these values with a polynomial of degree less than k . The latter encoding procedure, while being slightly less efficient, has the advantage that it gives rise to a [systematic code](#), that is, the original message is always contained as a subsequence of the codeword.

Simple encoding procedure: The message as a sequence of coefficients [\[edit\]](#)

In the original construction of [Reed & Solomon \(1960\)](#), the message $x = (x_1, \dots, x_k) \in F^k$ is mapped to the polynomial p_x with

$$p_x(a) = \sum_{i=1}^k x_i a^{i-1}.$$

The codeword of x is obtained by evaluating p_x at n different points $a_1, \dots, a_n$ of the field F . Thus the classical encoding function $C : F^k \rightarrow F^n$ for the Reed–Solomon code is defined as follows:

$$C(x) = (p_x(a_1), \dots, p_x(a_n)).$$

This function C is a [linear mapping](#), that is, it satisfies $C(x) = x \cdot A$ for the following $(k \times n)$ -matrix A with elements from F :

$$A = \begin{bmatrix} 1 & \dots & 1 & \dots & 1 \\ a_1 & \dots & a_k & \dots & a_n \\ a_1^2 & \dots & a_k^2 & \dots & a_n^2 \\ \vdots & & \vdots & & \vdots \\ a_1^{k-1} & \dots & a_k^{k-1} & \dots & a_n^{k-1} \end{bmatrix}$$

This matrix is the transpose of a [Vandermonde matrix](#) over F . In other words, the Reed–Solomon code is a [linear code](#), and in the classical encoding procedure, its [generator matrix](#) is A .

Systematic encoding procedure: The message as an initial sequence of values [\[edit\]](#)

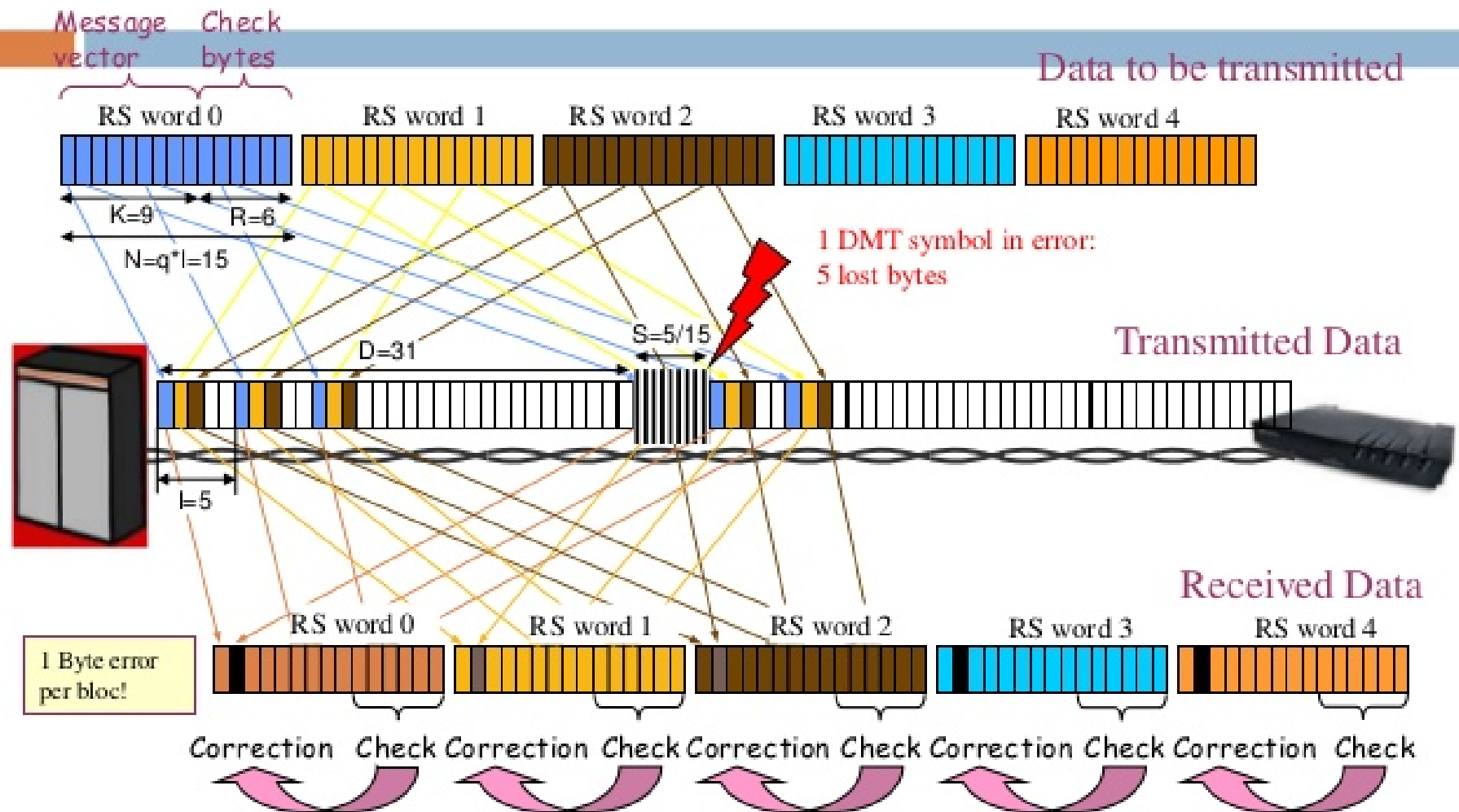
There is an alternative encoding procedure that also produces the Reed–Solomon code, but that does so in a [systematic](#) way. Here, the mapping from the message x to the polynomial p_x works differently: the polynomial p_x is now defined as the unique polynomial of degree less than k such that

$$p_x(a_i) = x_i \text{ holds for all } i \in \{1, \dots, k\}.$$

To compute this polynomial p_x from x , one can use [Lagrange interpolation](#). Once it has been found, it is evaluated at the other points $a_{k+1}, \dots, a_n$ of the field. The alternative encoding function $C : F^k \rightarrow F^n$ for the Reed–Solomon code is then again just the sequence of values:

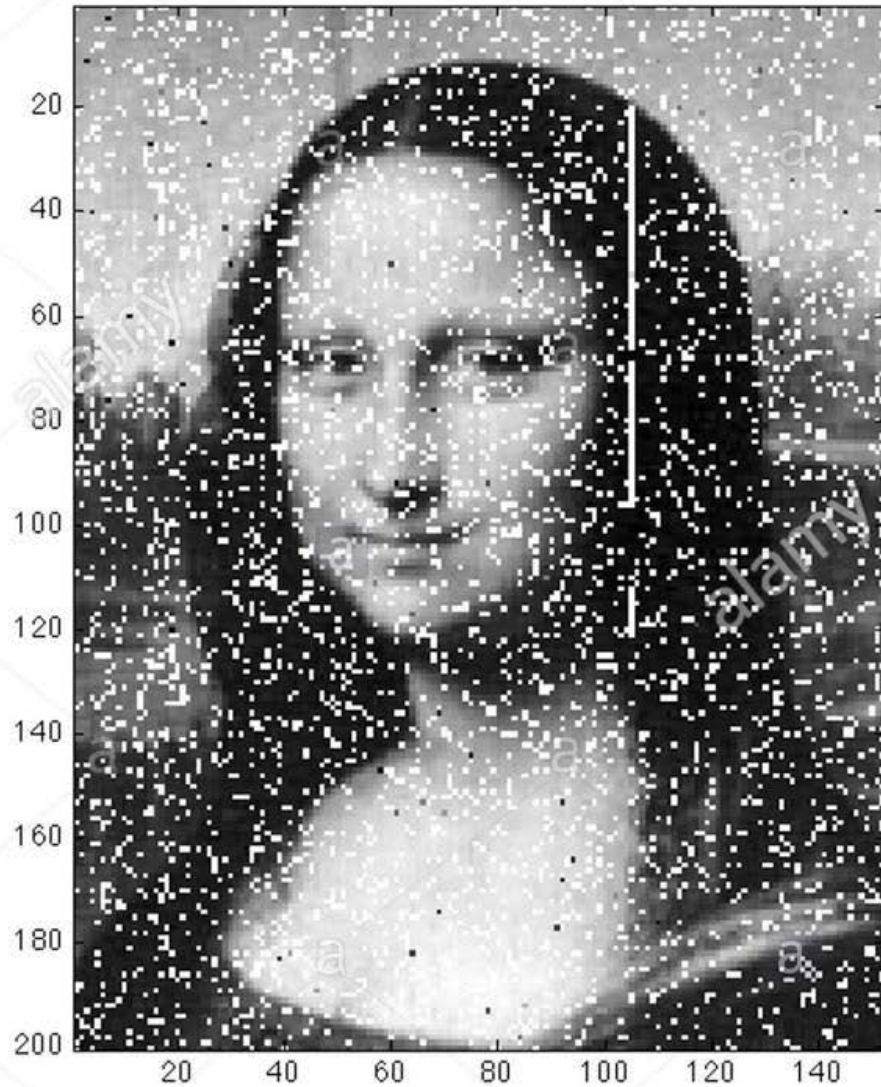
$$C(x) = (p_x(a_1), \dots, p_x(a_n)).$$

Reed-Solomon coding (interleaving)

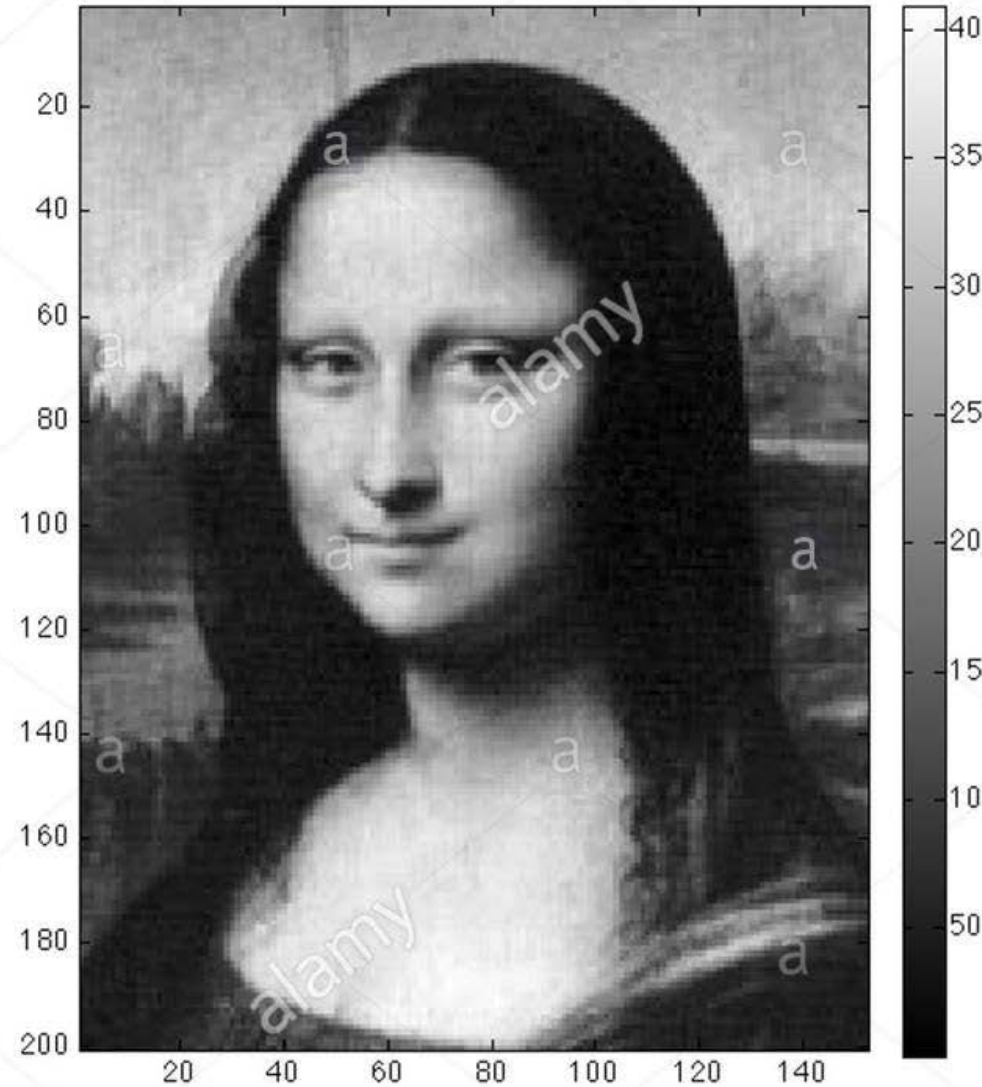




No Coding



Rate 2/3 Reed-Solomon Coding



***Köszönöm megtisztelő
figyelmüket!***

