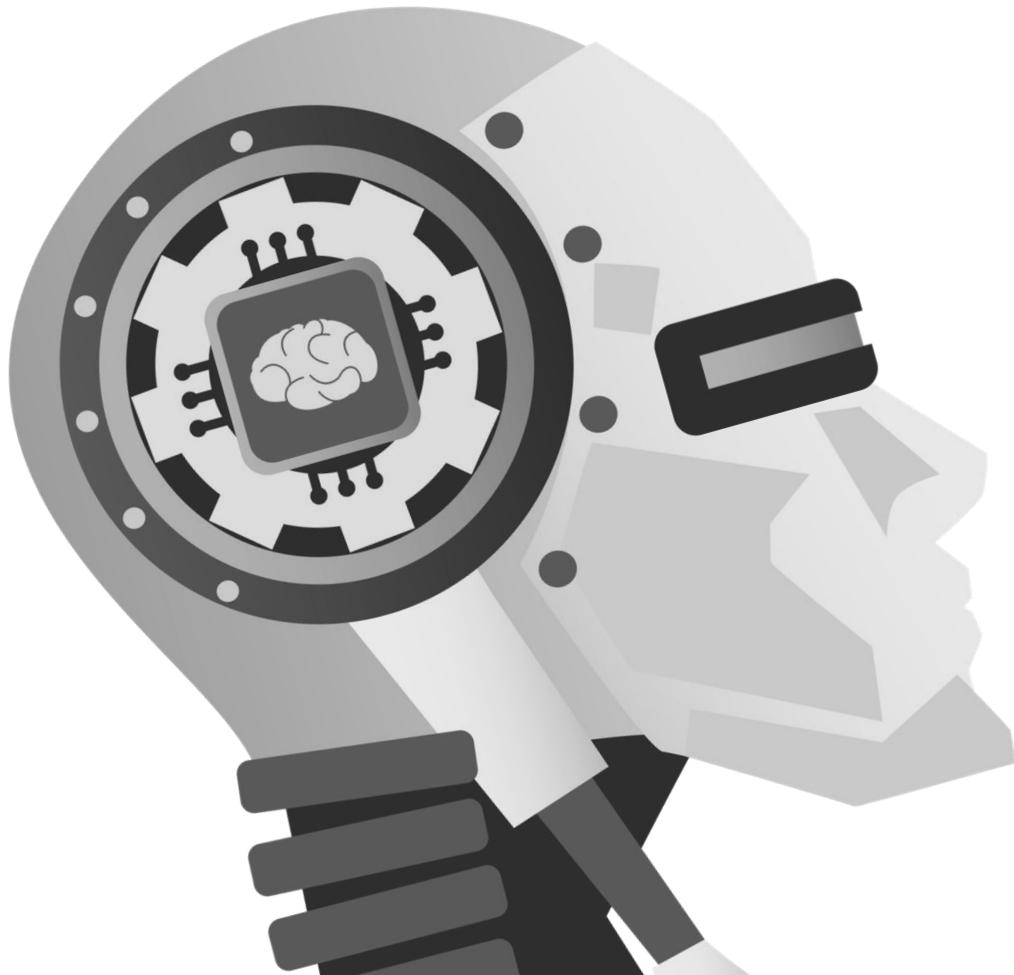


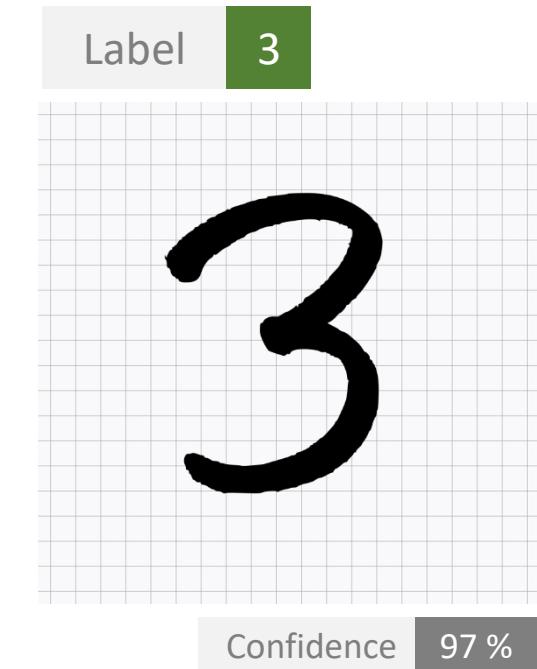
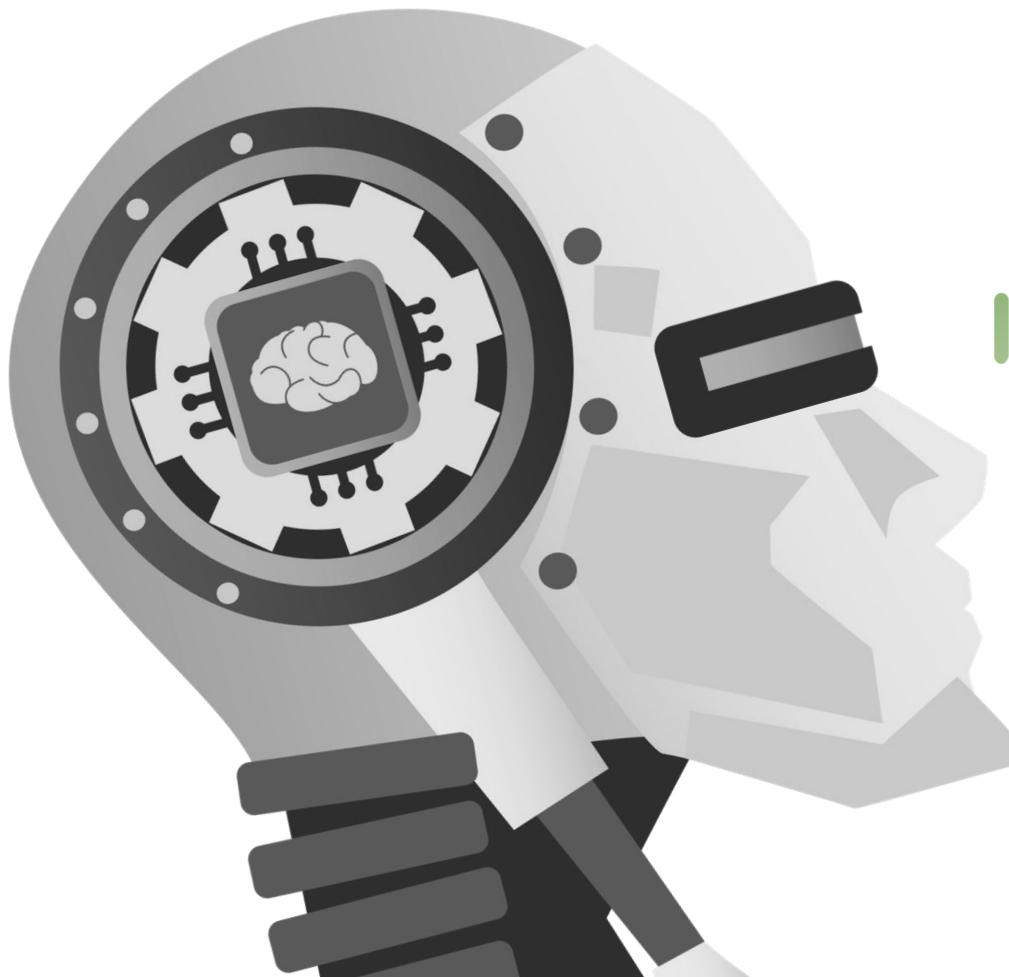
# Programovanie 2

## Zadanie 2



# Programovanie 2

## Zadanie 2



# Obsah

- Odovzdávanie/podmienky
- Vysvetlenie a úlohy
- Bodovanie
- Živá ukážka
  - Skompilovanie projektu
  - Použitie programu v termináli
  - Rozpoznanie rukou písanej číslice

# Odvzdávanie/podmienky

- **Deadline:** 19. 3. 2024, 10:00:00
- 10 bodov
- Anti-plagiátorský systém



[www.prog2.dev](http://www.prog2.dev)

# Odvzdávanie/podmienky

## Penalizácia

- |                |                  |
|----------------|------------------|
| 1. pokus ..... | <b>max. 10 b</b> |
| 2. pokus ..... | max. 9 b         |
| 3. pokus ..... | max. 7 b         |
| 4. pokus ..... | max. 5 b         |
| 5. pokus ..... | max. 3 b         |
| 6. pokus ..... | max. 1 b         |

## Zadanie 2

- Cieľom zadania je napísať program v jazyku C na klasifikáciu obrázkov z databázy MNIST pomocou neurónovej siete.
- MNIST obsahuje rukou písané čísllice 0 až 9.



# Zadanie 2

- Inšpirácia existujúcou prácou **Andrew Carter: MNIST Neural Network in C.**
- GitHub repozitár:

[https://github.com/AndrewCarterUK/  
mnist-neural-network-plain-c](https://github.com/AndrewCarterUK/mnist-neural-network-plain-c)



# MNIST Neural Network in C

This source code seeks to replicate the (now removed) [MNIST For ML Beginners](#) tutorial from the Tensorflow website using plain C code.

The task is to recognise digits, such as the ones below, as accurately as possible.



By [AndrewCarterUK](#)

## Contents

- [mnist.c](#): Glue code that runs the algorithm steps and reports algorithm accuracy
- [mnist\\_file.c](#): Retrieves images and labels from the MNIST dataset
- [neural\\_network.c](#): Implements training and prediction routines for a simple neural network

## Usage

```
make  
../mnist
```



## Zadanie 2

V zadaní si **precvičíte**:

- matematické operácie,
- funkcie,
- polia,
- prácu s projektom, ktorý je zložený z viacerých hlavičkových a zdrojových súborov

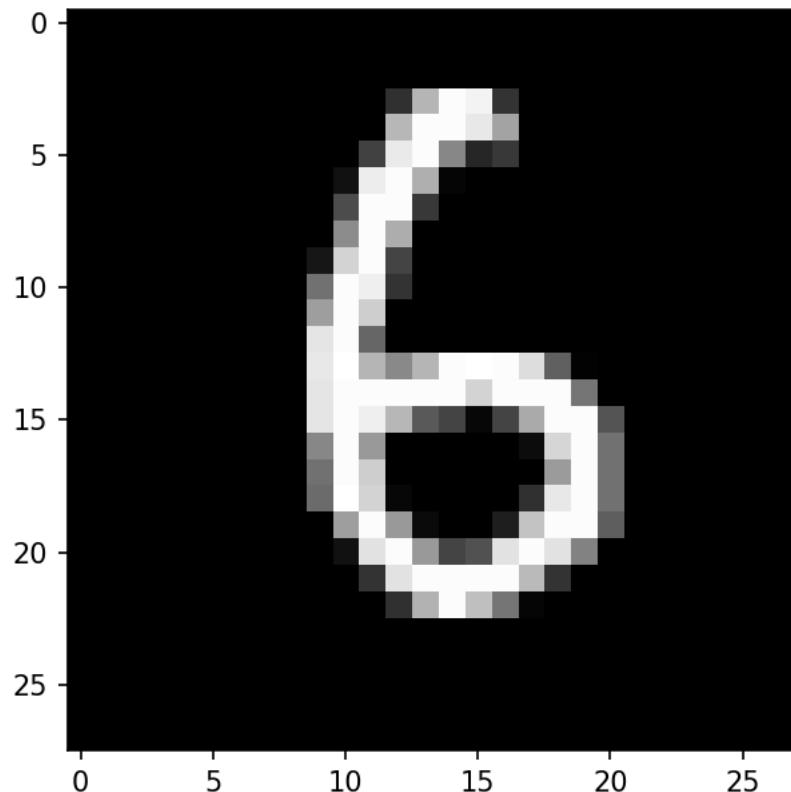
# MNIST databáza obrázkov

- Rozsiahla zbierka obrázkov rukou písaných číslíc.
- Obsahuje číslice 0-9.
- Jedná sa o obrovskú databázu:
  - 60 000 trénovacích obrázkov
  - 10 000 testovacích obrázkov

# MNIST databáza obrázkov

- Rozmer každého obrázku je  $28 \times 28$  pixelov.
- Každý obrázok je reprezentovaný odtieňmi šedej farby (tzv. grayscale), t.j. pixely sú v rozsahu  $\langle 0,255 \rangle$ .

28 px



28 px



# MNIST databáza obrázkov

- Obrázok budeme v zadaní reprezentovať pomocou 1D poľa (prepísané pixely po riadkoch zľava doprava).

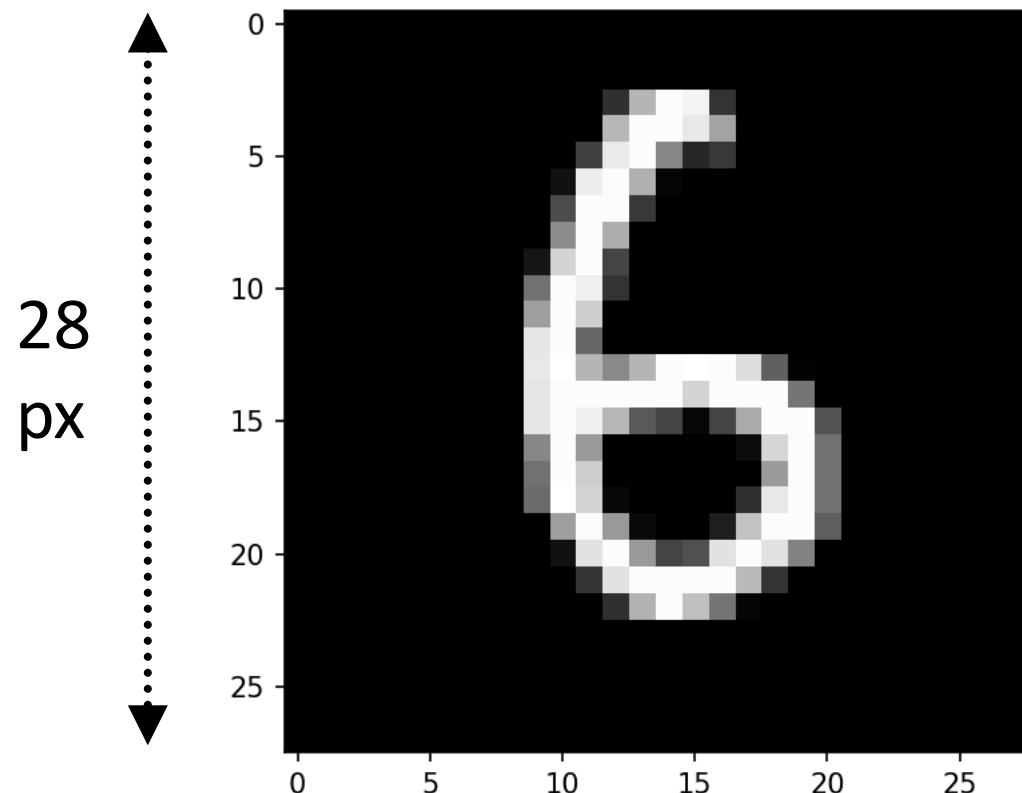
# MNIST databáza obrázkov

- Obrázok budeme v zadaní reprezentovať pomocou 1D poľa (prepísané pixely po riadkoch zľava doprava).
- Obrázky sú uložené v TXT súboroch a načítajú sa do poľa pomocou presmerovania štandardného vstupu (stdin).

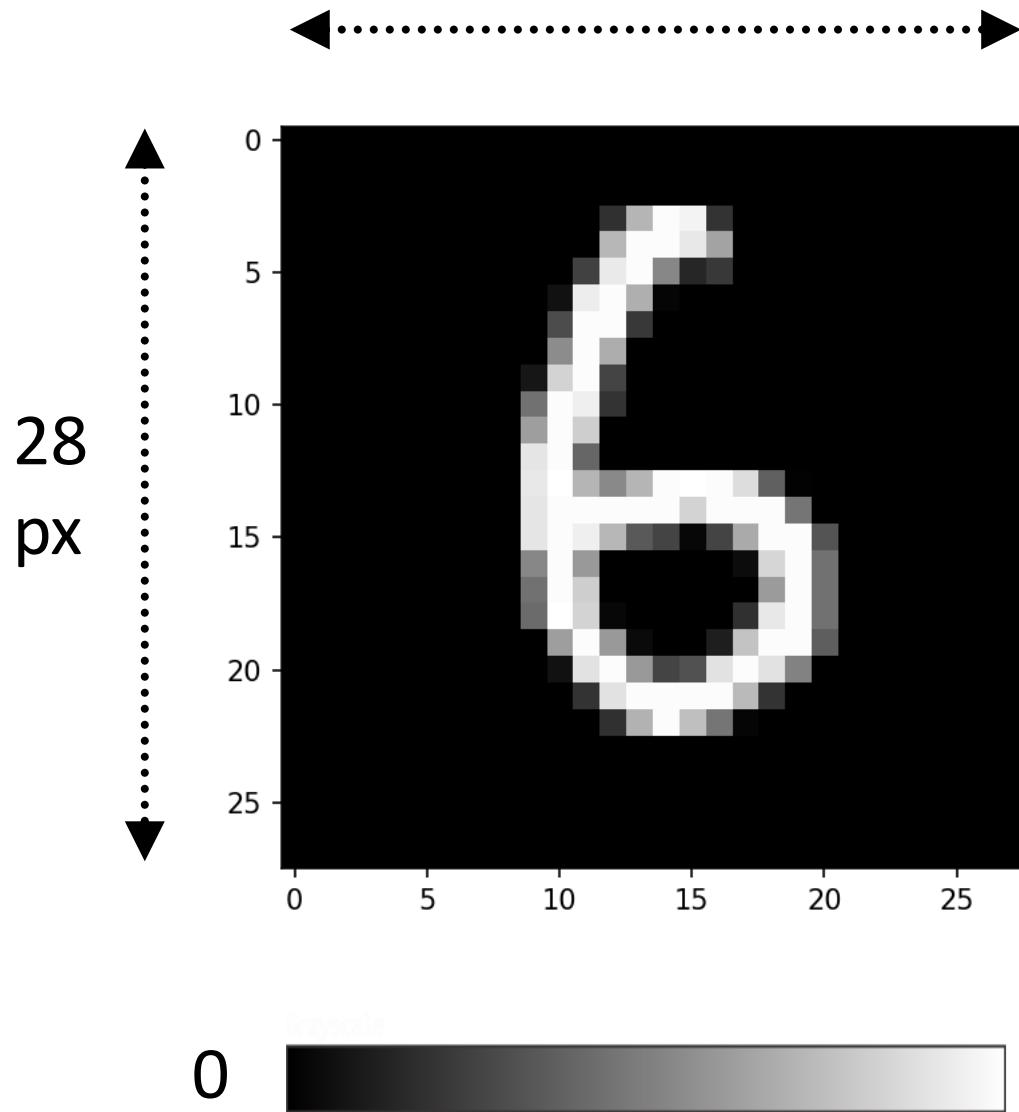
# MNIST databáza obrázkov

- Obrázok budeme v zadaní reprezentovať pomocou 1D poľa (prepísané pixely po riadkoch zľava doprava).
- Obrázky sú uložené v TXT súboroch a načítajú sa do poľa pomocou presmerovania štandardného vstupu (stdin).
- Hodnoty pixelov obrázku sú normalizované z pôvodného rozsahu  $\langle 0,255 \rangle$  do rozsahu  $\langle 0,1 \rangle$  (je to vhodnejšia reprezentácia pre NS).

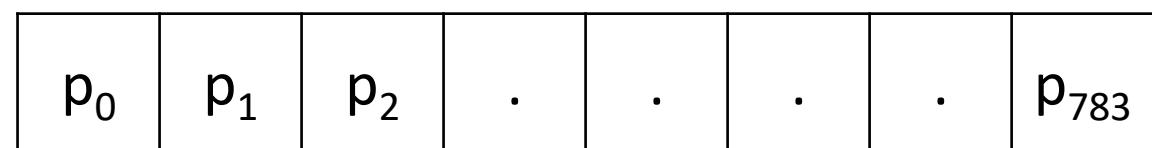
28 px



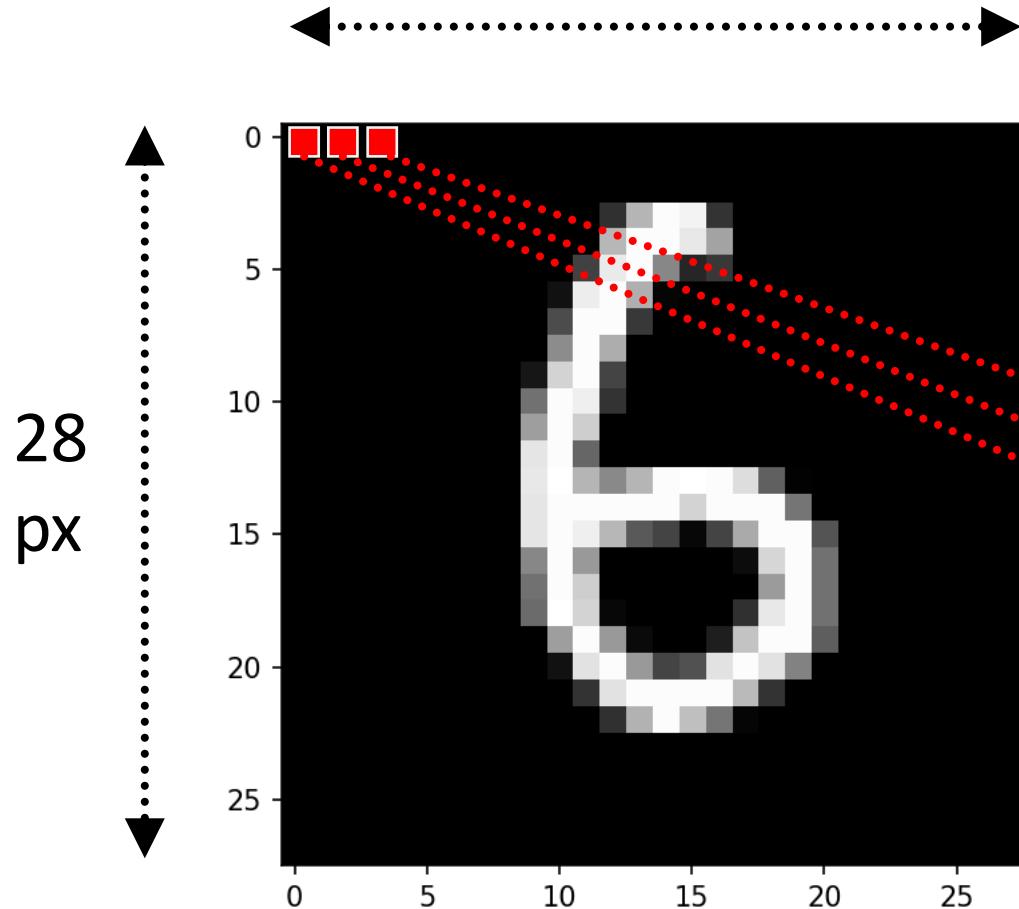
28 px



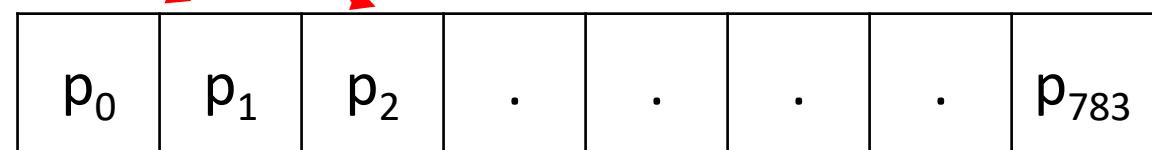
1D pole na  
reprezentáciu obrázka  
(typ double)



28 px

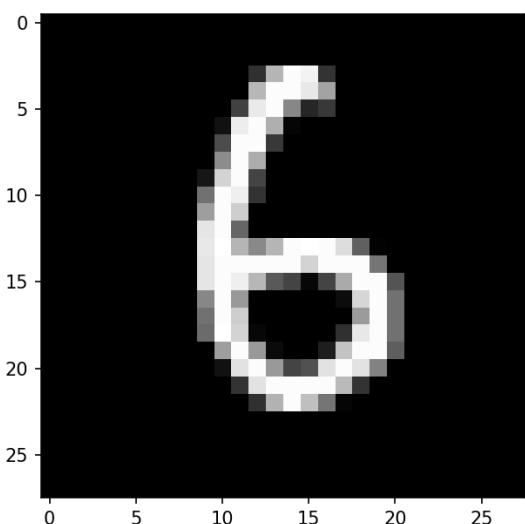


1D pole na  
reprezentáciu obrázka  
(typ double)



0 255

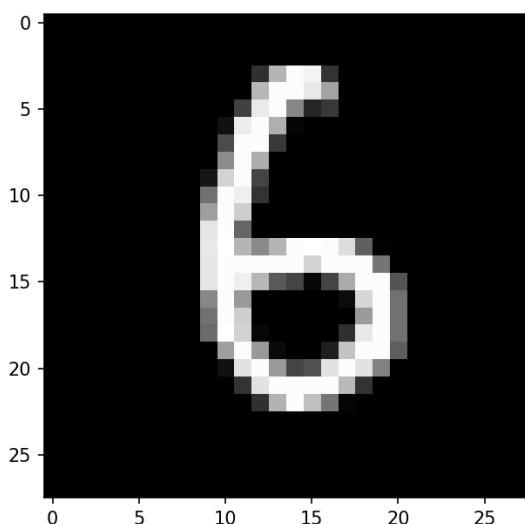
## Príklad obrázku uloženého v TXT súbore.



example.txt X

```
1  0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00  
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00  
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00  
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00  
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00  
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00  
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00  
0.00 0.00 0.00 0.00 0.14 0.49 0.75 0.85 1.00 1.00 1.00 1.00 0.95 0.20  
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00  
0.98 0.92 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00  
0.99 0.88 0.56 0.19 0.05 0.05 0.05 0.41 0.99 0.99 0.44 0.00 0.00 0.00  
0.00 0.00 0.00 0.00 0.37 0.88 0.99 0.65 0.44 0.05 0.00 0.00 0.00 0.00  
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.36 0.93  
0.00 0.00 0.00 0.00 0.59 0.99 0.46 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00  
0.85 0.19 0.02 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.59  
0.00 0.00 0.00 0.00 0.00 0.00 0.44 0.99 0.44 0.00 0.00 0.00 0.00 0.00 0.00  
0.72 0.95 0.07 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00  
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.26 0.99 0.94 0.00 0.00 0.00 0.00 0.00  
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00  
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00  
0.00 0.17 0.62 0.99 0.52 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
```

Príklad obrázku  
uloženého v TXT  
súbore.





example.txt



1 riadok obsahujúci 784 hodnôt pixelov v rozsahu  $\langle 0,1 \rangle$ .  
Hodnoty sú oddelené medzerou.

# MNIST databáza obrázkov

- Viac detailov na Kaggle:

<https://www.kaggle.com/datasets/hojjatk/mnist-dataset>



+ Create

Home

Competitions

Datasets

Models

Code

Discussions

Learn

More



Search



HOJJAT KHODABAKHSH · UPDATED 5 YEARS AGO

# MNIST Dataset

The MNIST database of handwritten digits (<http://yann.lecun.com>)

Data Card

Code (112)

Discussion (0)

Suggestions (0)

## About Dataset

### Context

MNIST is a subset of a larger set available from NIST (it's copied from <http://yann.lecun.com>)

### Content

The MNIST database of handwritten digits has a training set of 60,000 examples, and a test set of 10,000 examples.

Four files are available:

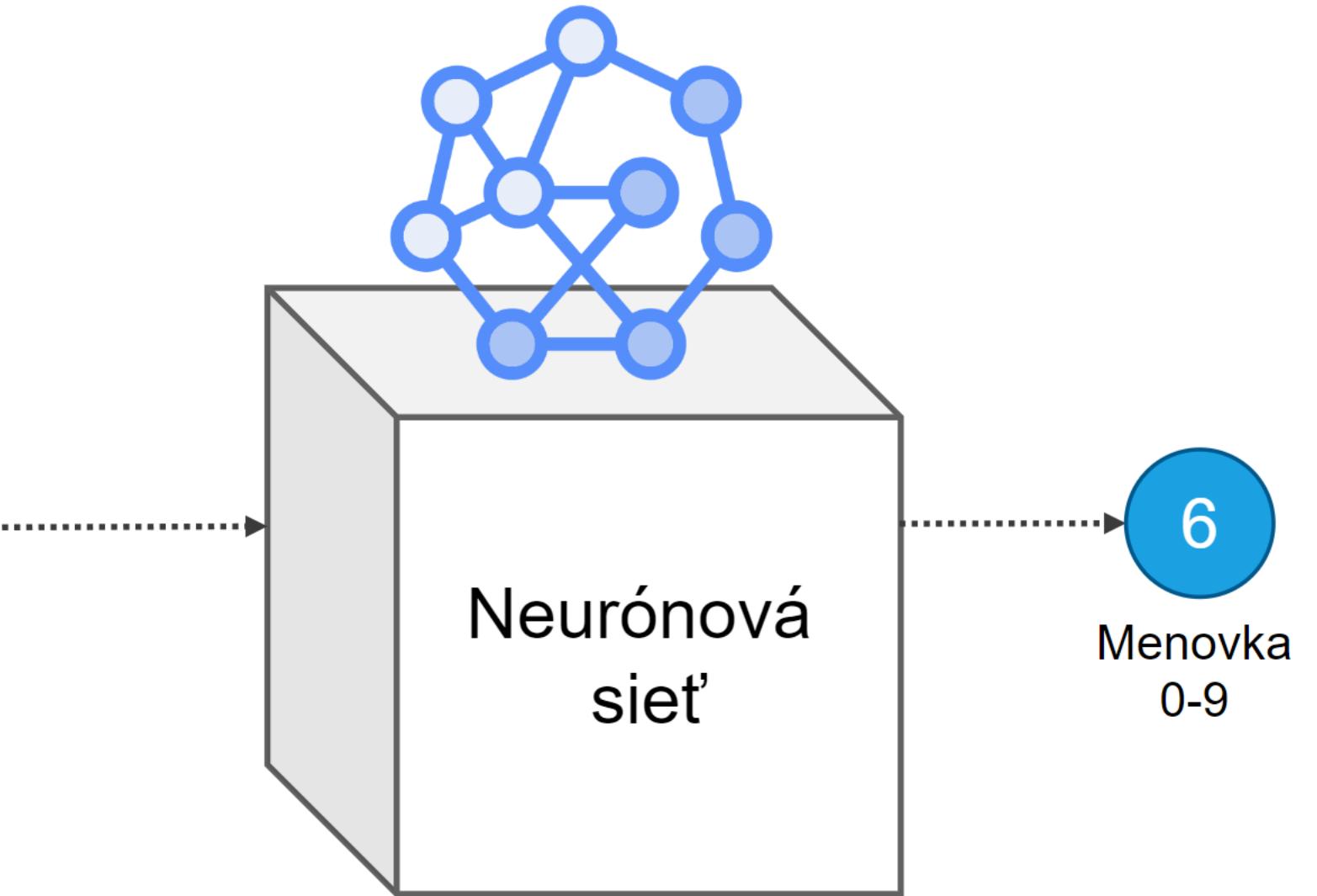
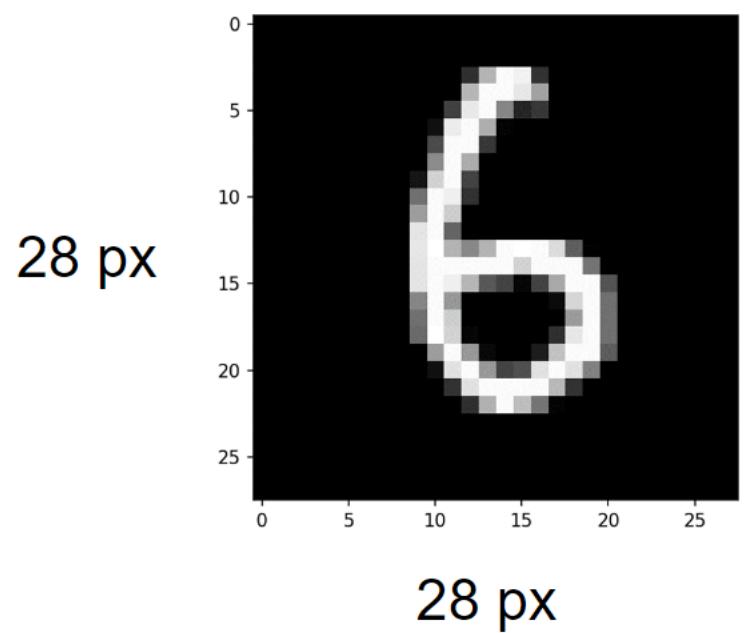
- train-images-idx3-ubyte.gz: training set images (9912422 bytes)

# Neurónová siet' (black box)

- Na rozpoznávanie číslíc sa použije **1-vrstvová neurónová siet' (NS)**.
- **Vstup NS**: obrázok (1D pole pixelov).
- **Výstup NS**: klasifikácia obrázku, t. j. určenie triedy, do ktorej obrázok patrí (číslo 0 až 9).

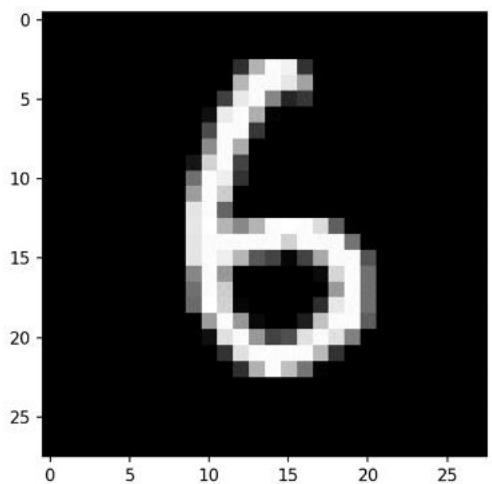
# Neurónová siet' (black box)

- **Zjednodušene:** NS je algoritmus, ktorý je natrénovaný na riešenie komplexného nelineárneho problému  $X \Rightarrow Y$ .
- V našom prípade je  $X$  obrázok (pole pixelov) a  $Y$  je menovka obrázku (číslo 0-9).



# Architektúra neurónovej siete

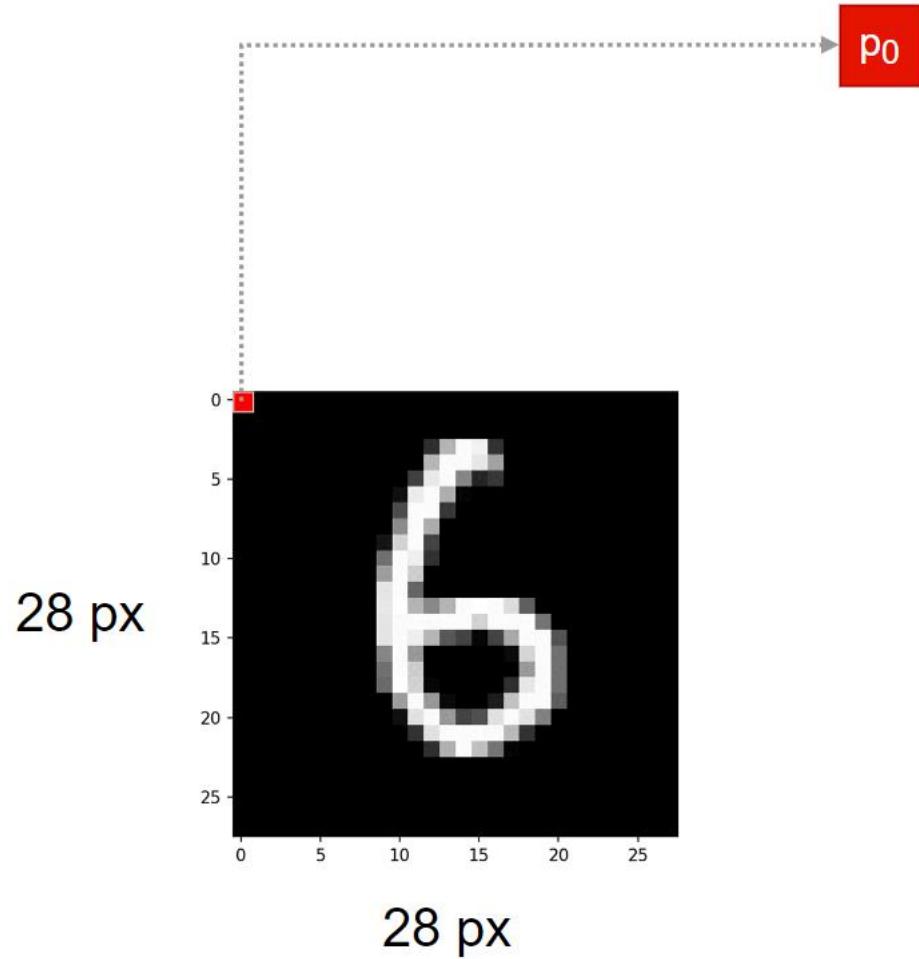
- NS v tomto zadaní bude obsahovať **1 vrstvu** neurónov.
- Počet neurónov je **10** (protože klasifikujeme 10 rôznych typov číslic).
- Všetky neuróny použijú **aktivačnú funkciu ReLU**.



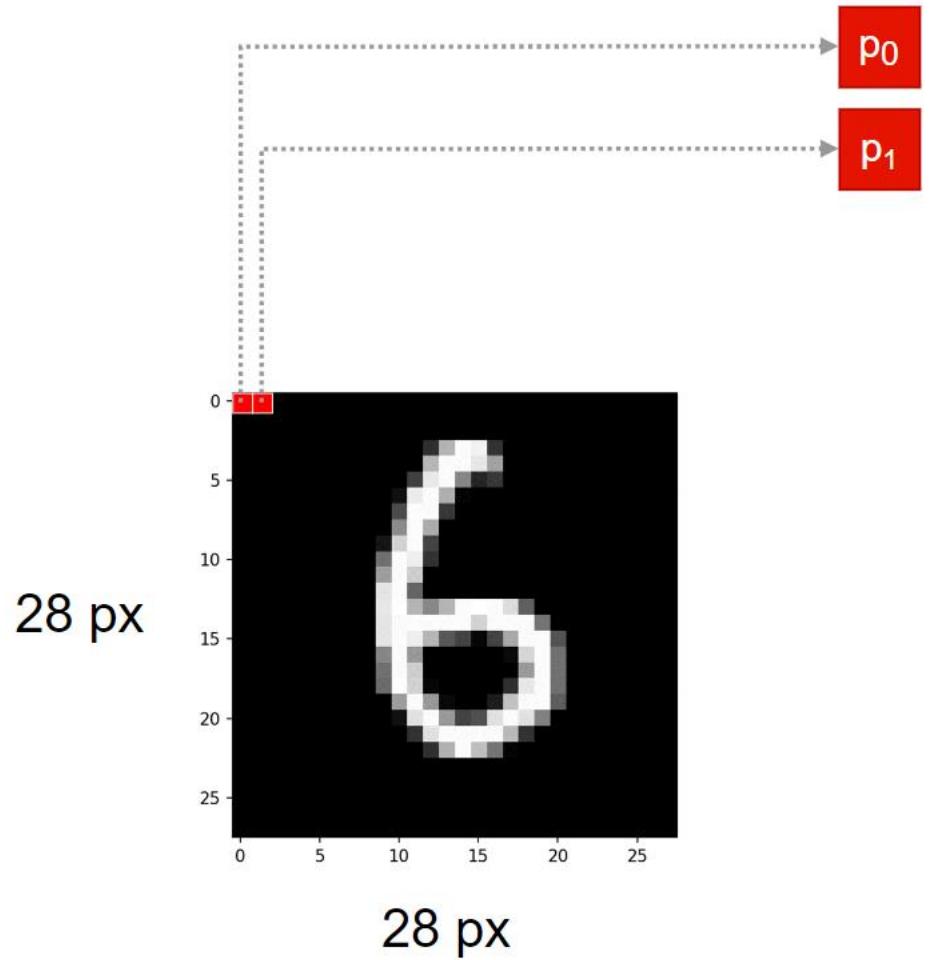
28 px

28 px

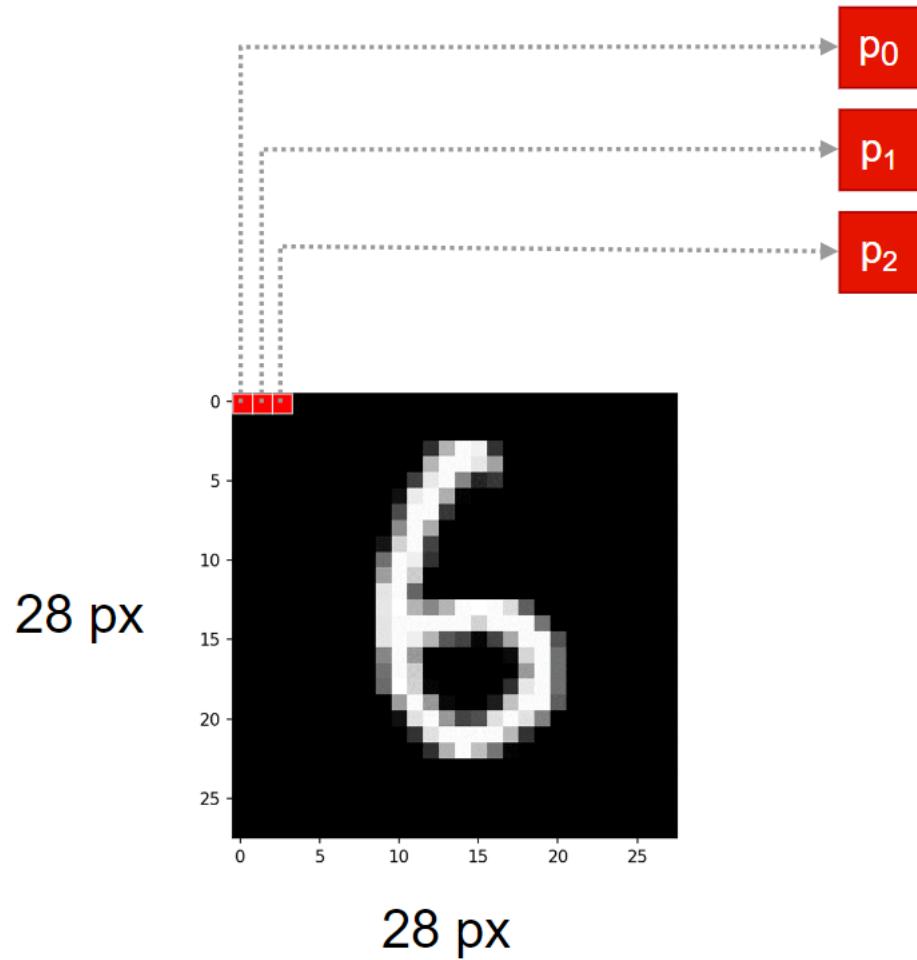
# Pixely obrázku



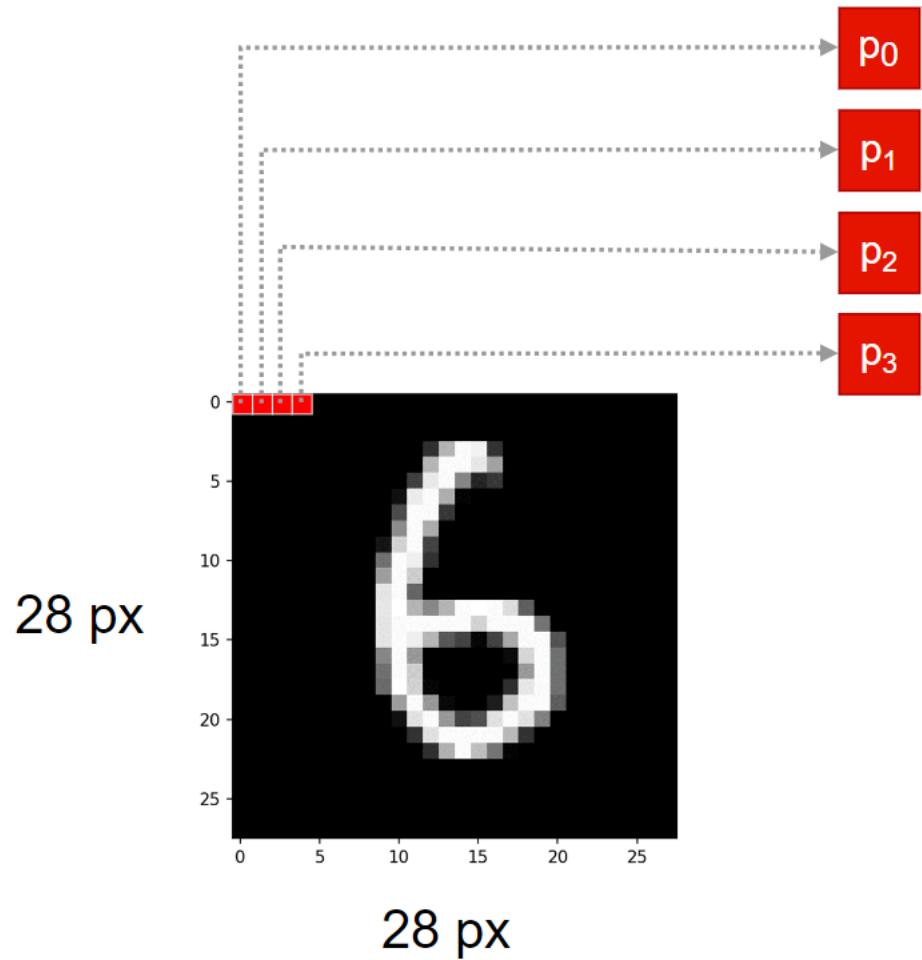
## Pixely obrázku



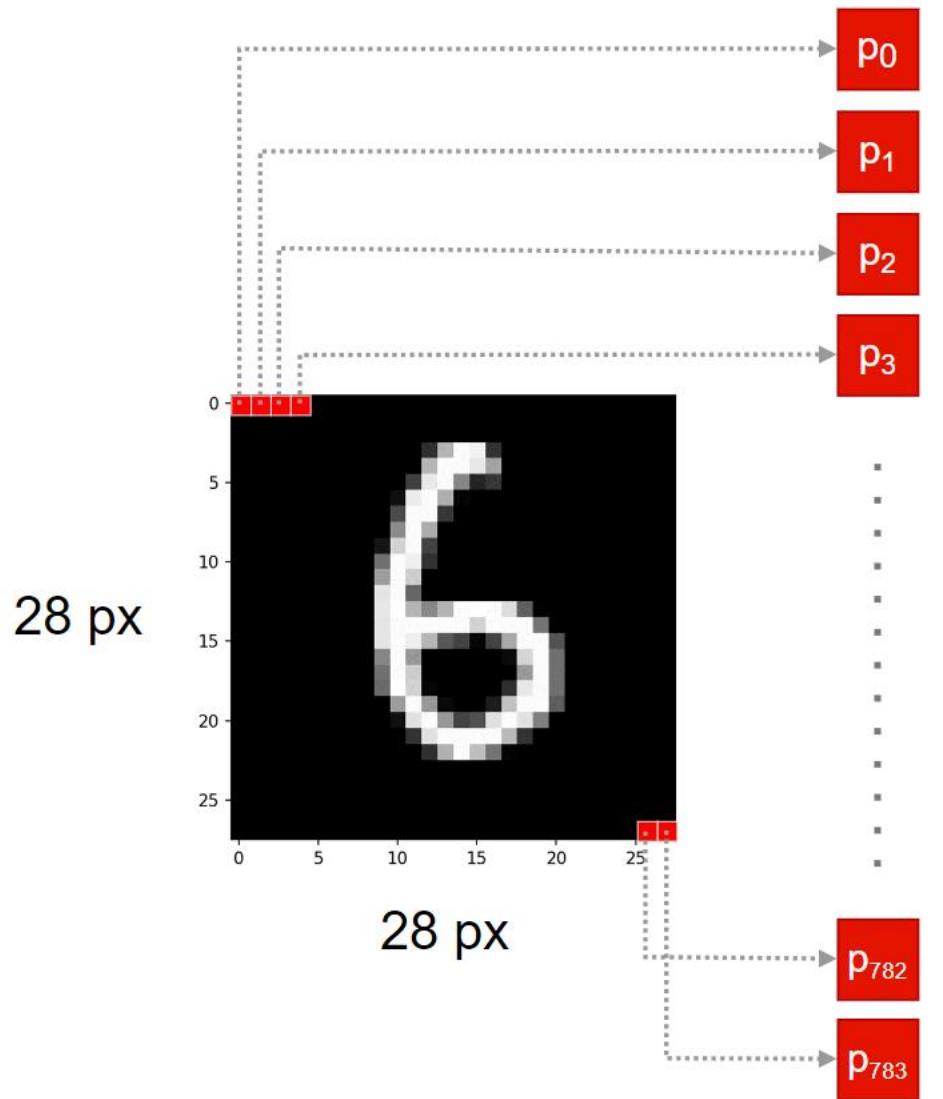
## Pixely obrázku

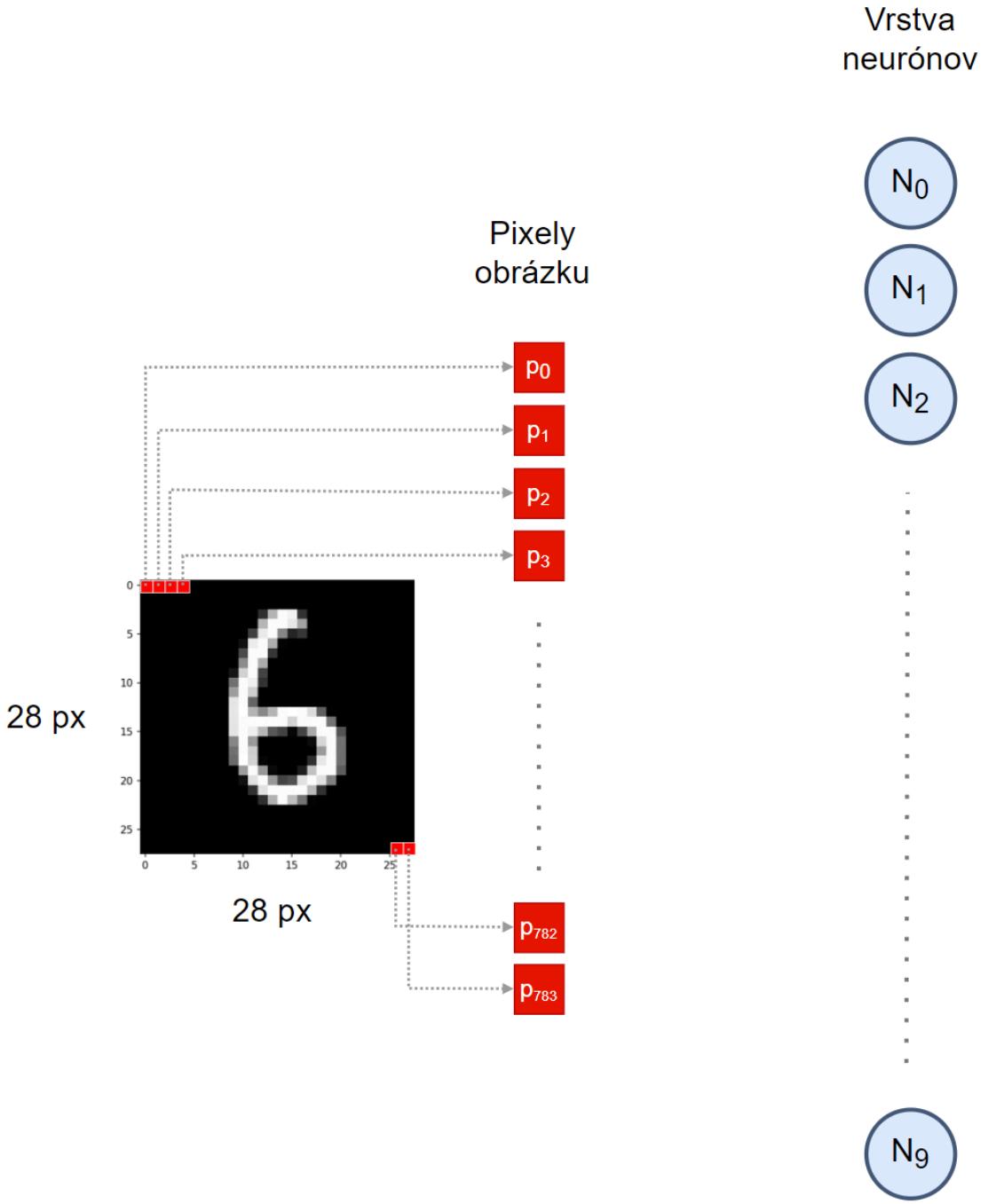


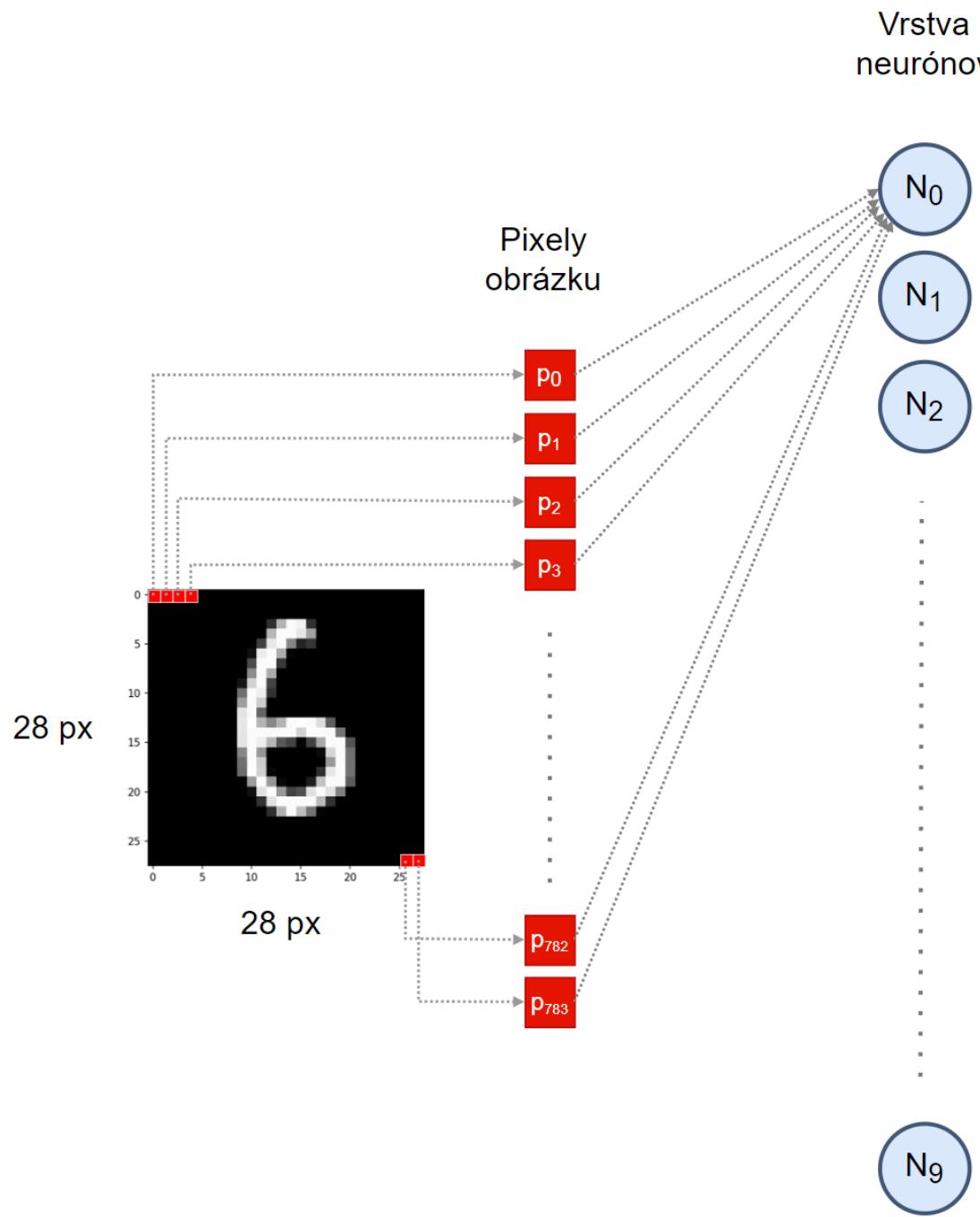
## Pixely obrázku

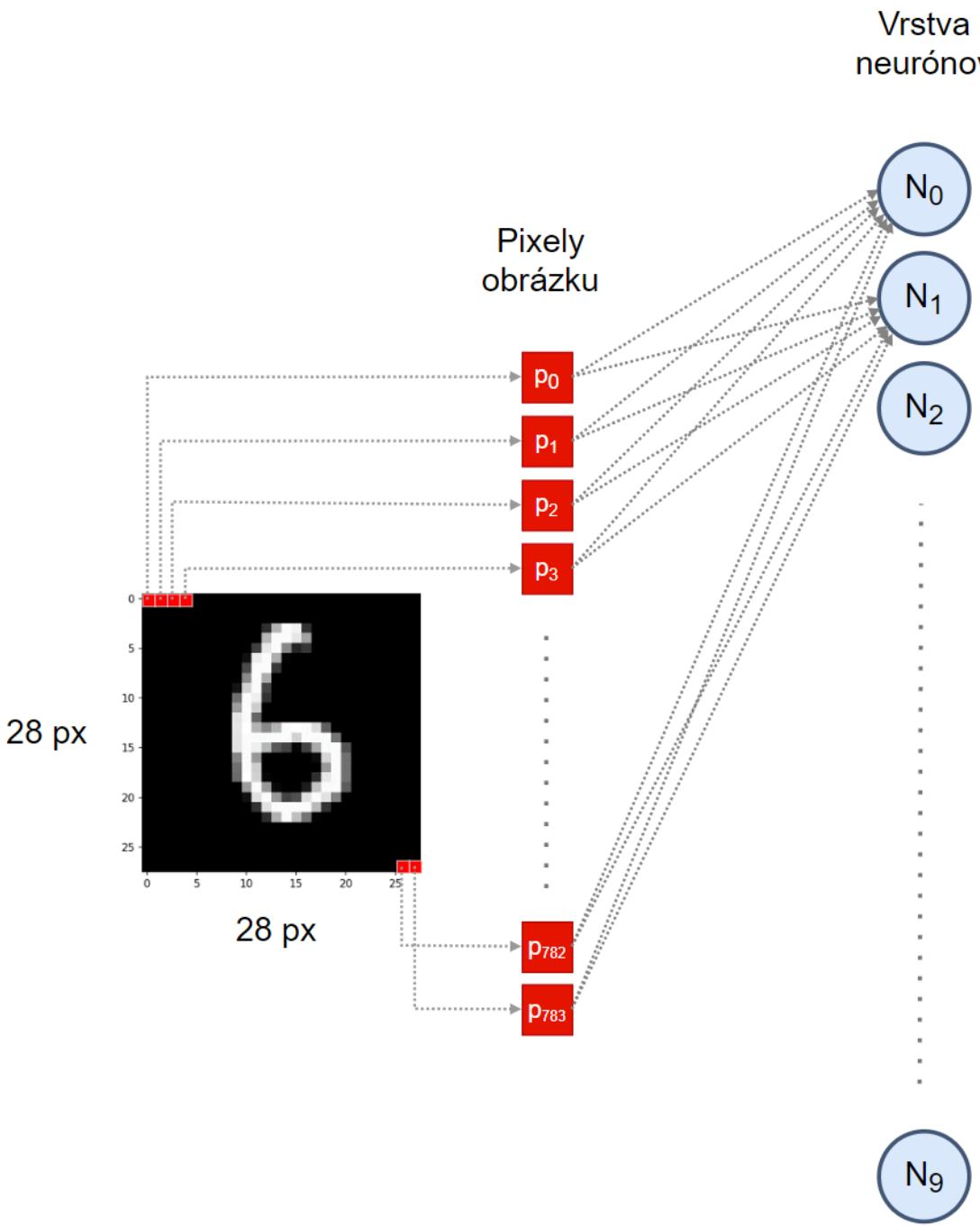


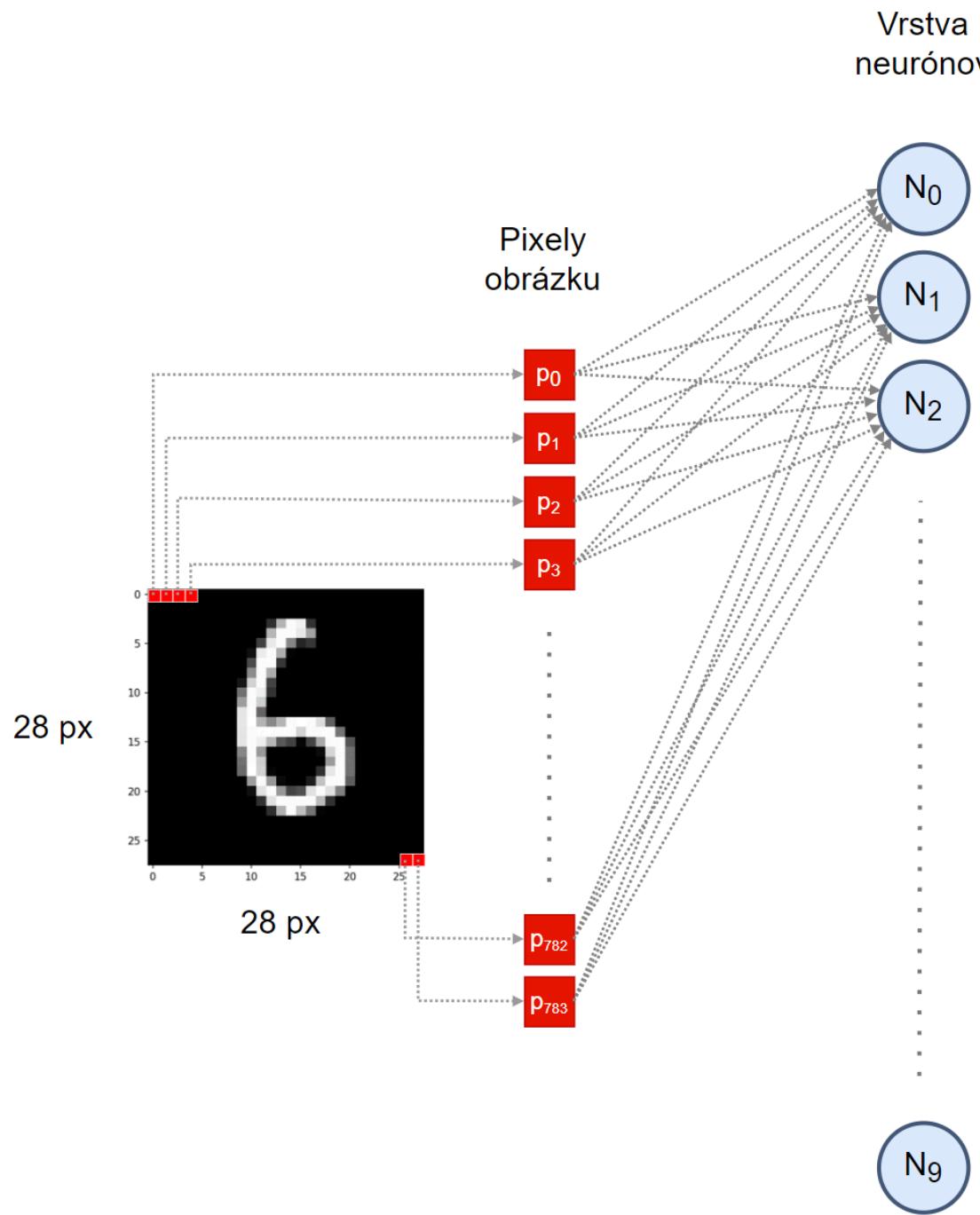
## Pixely obrázku

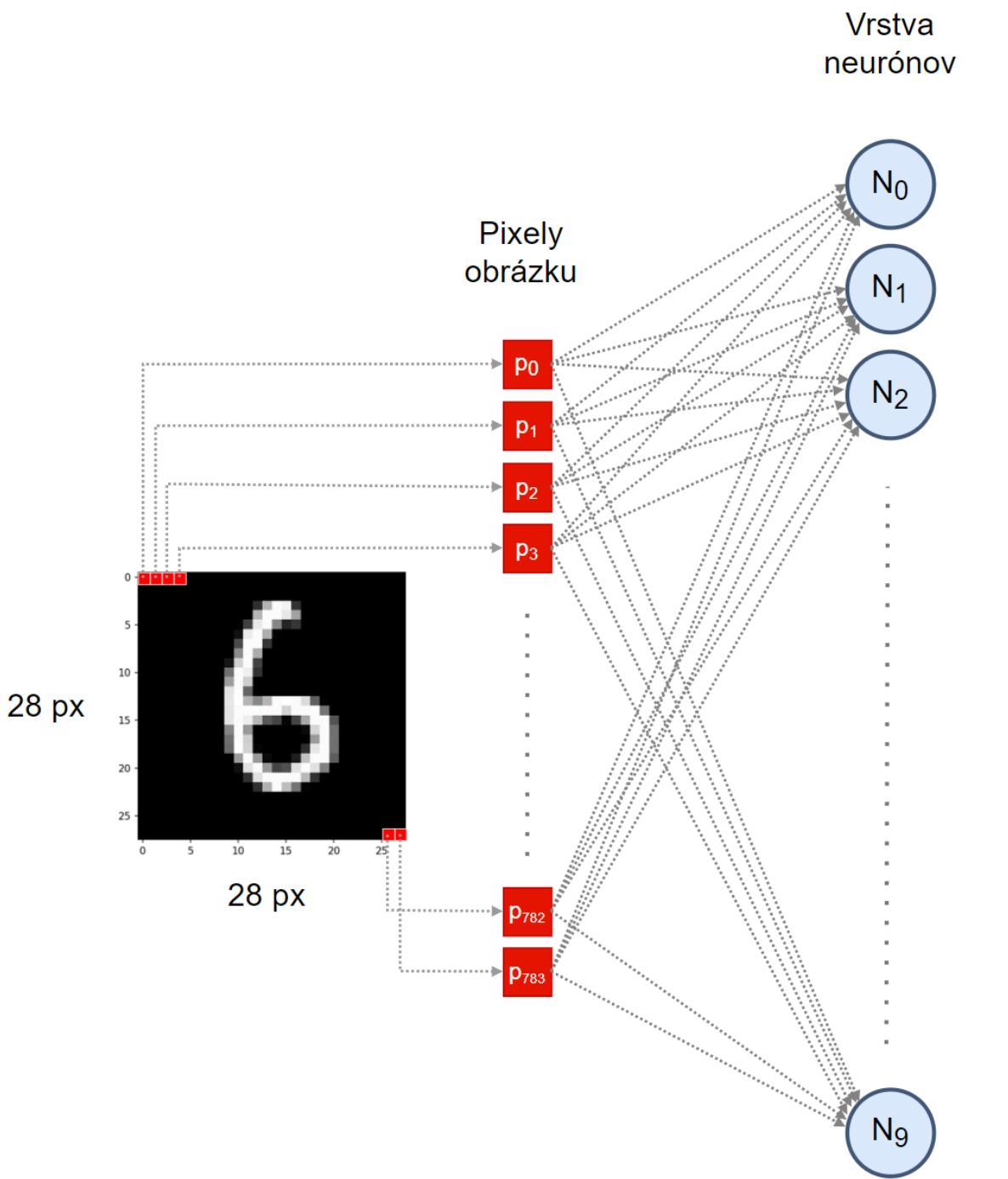


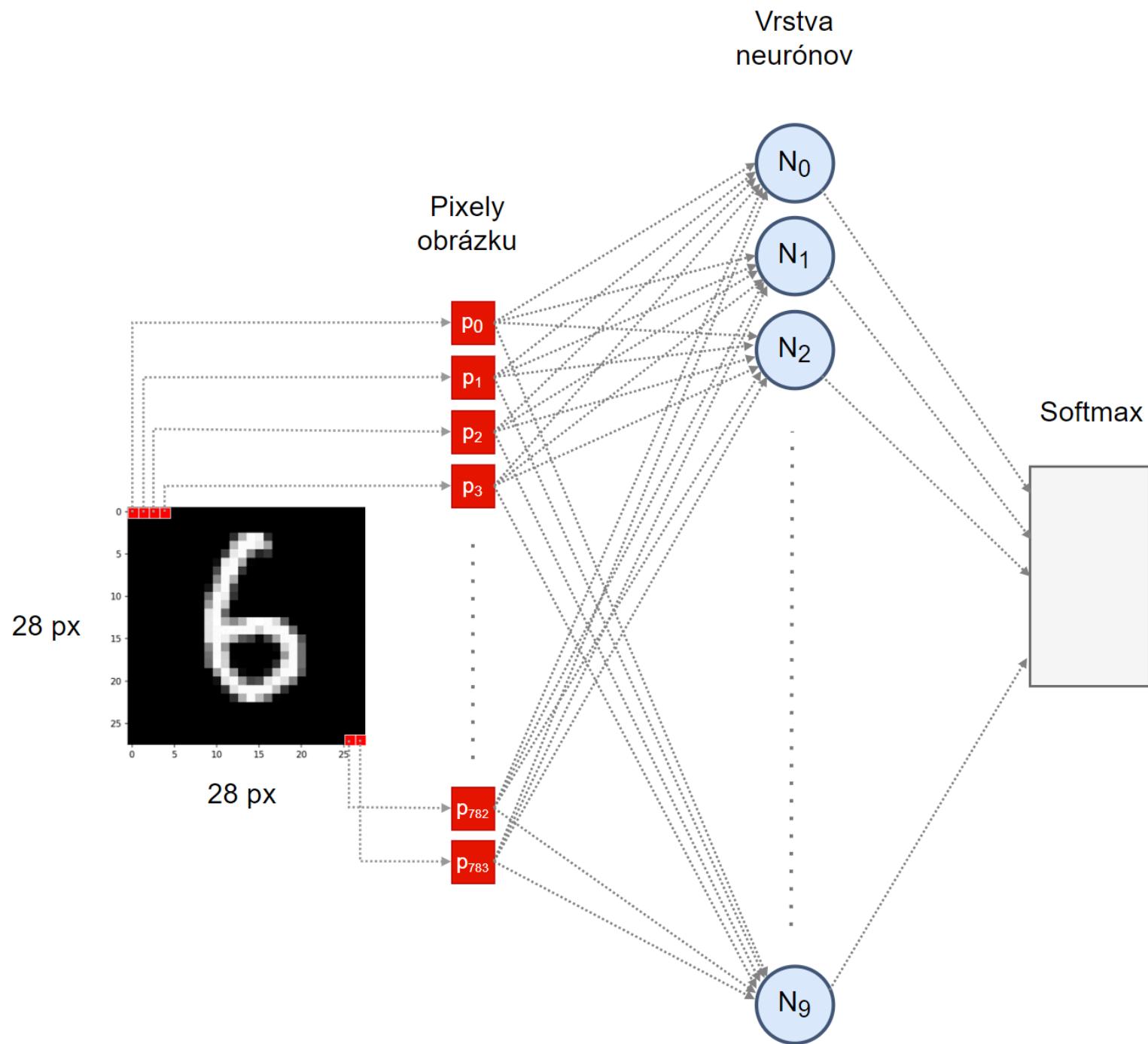


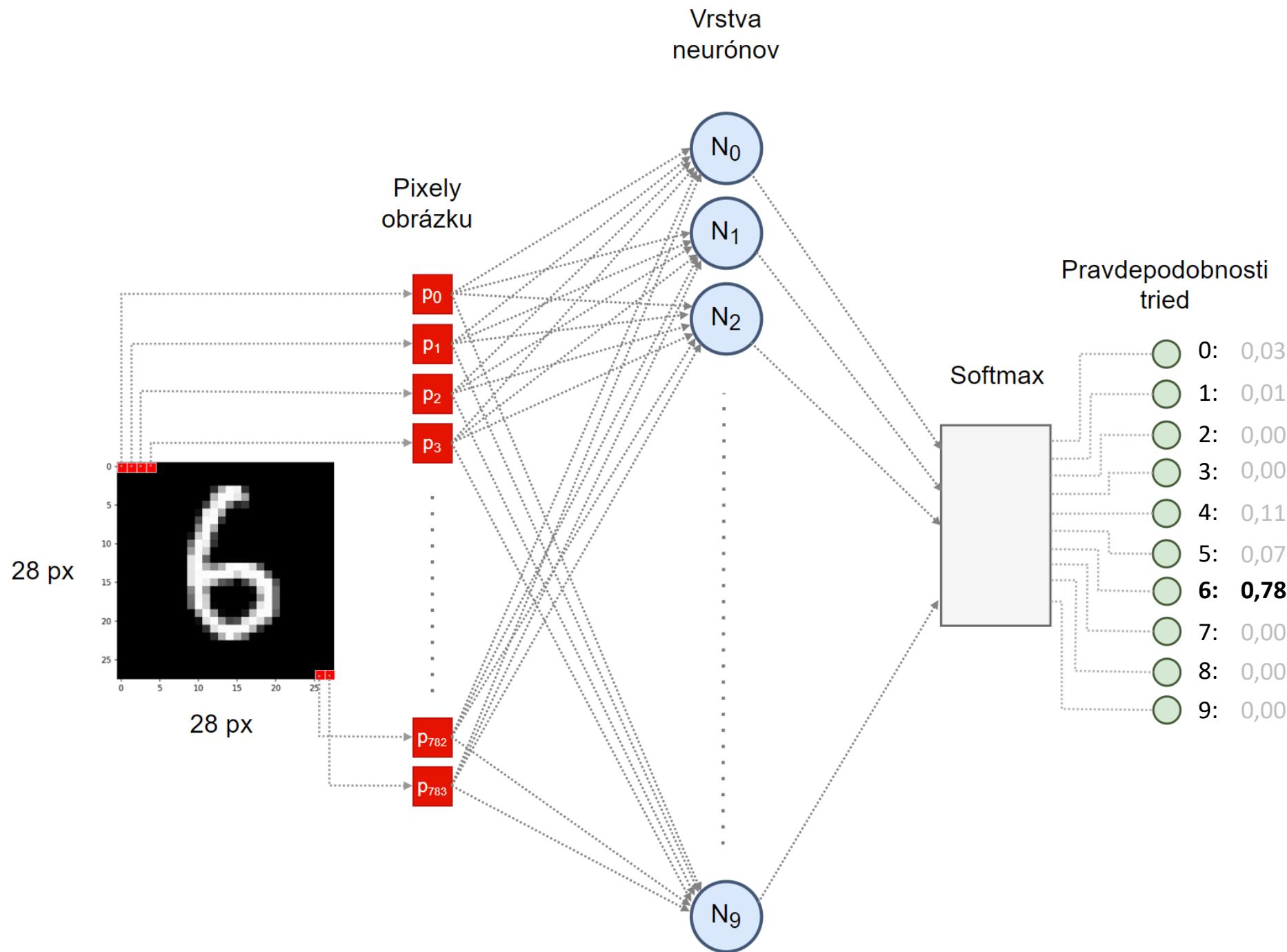


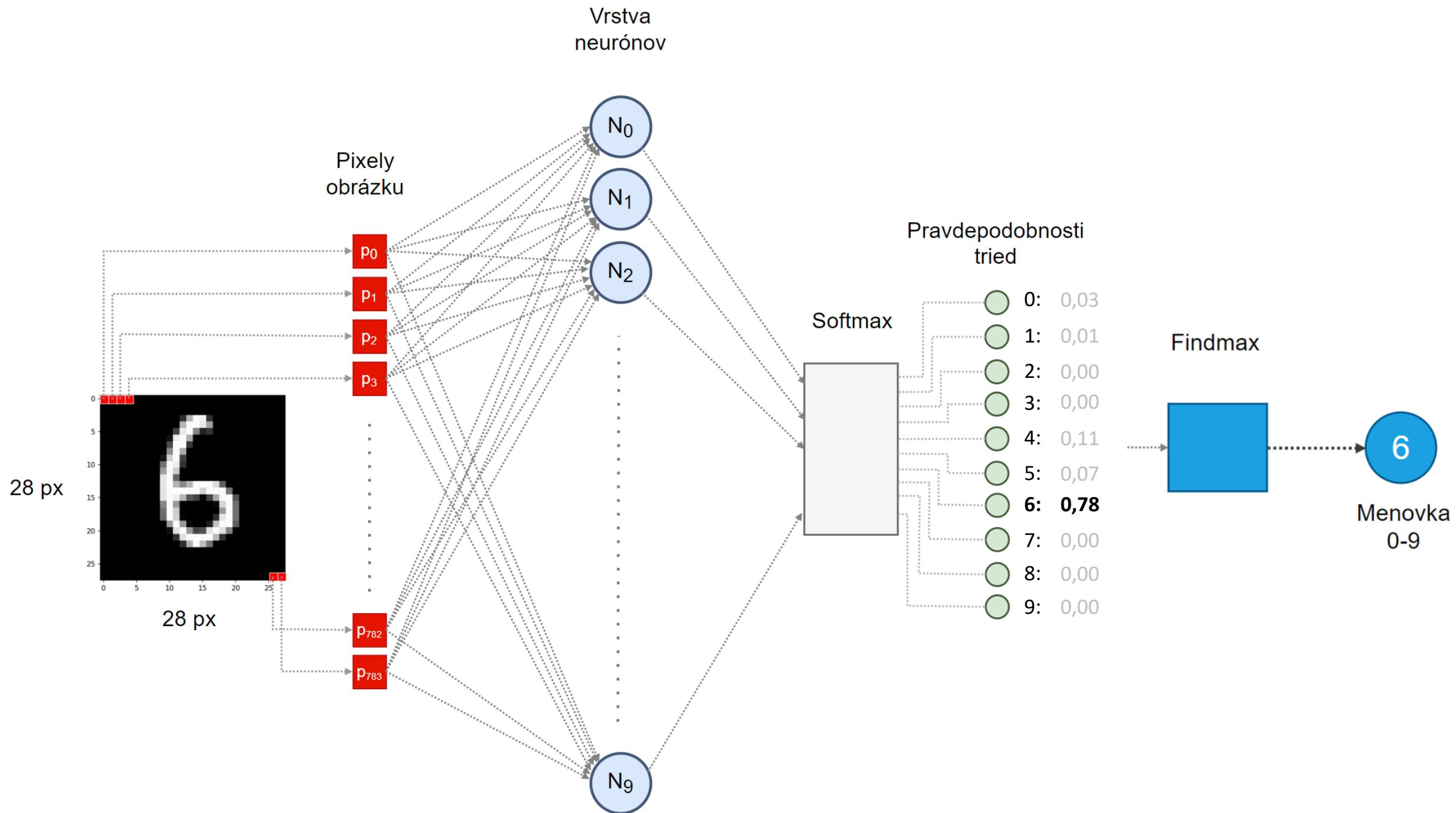




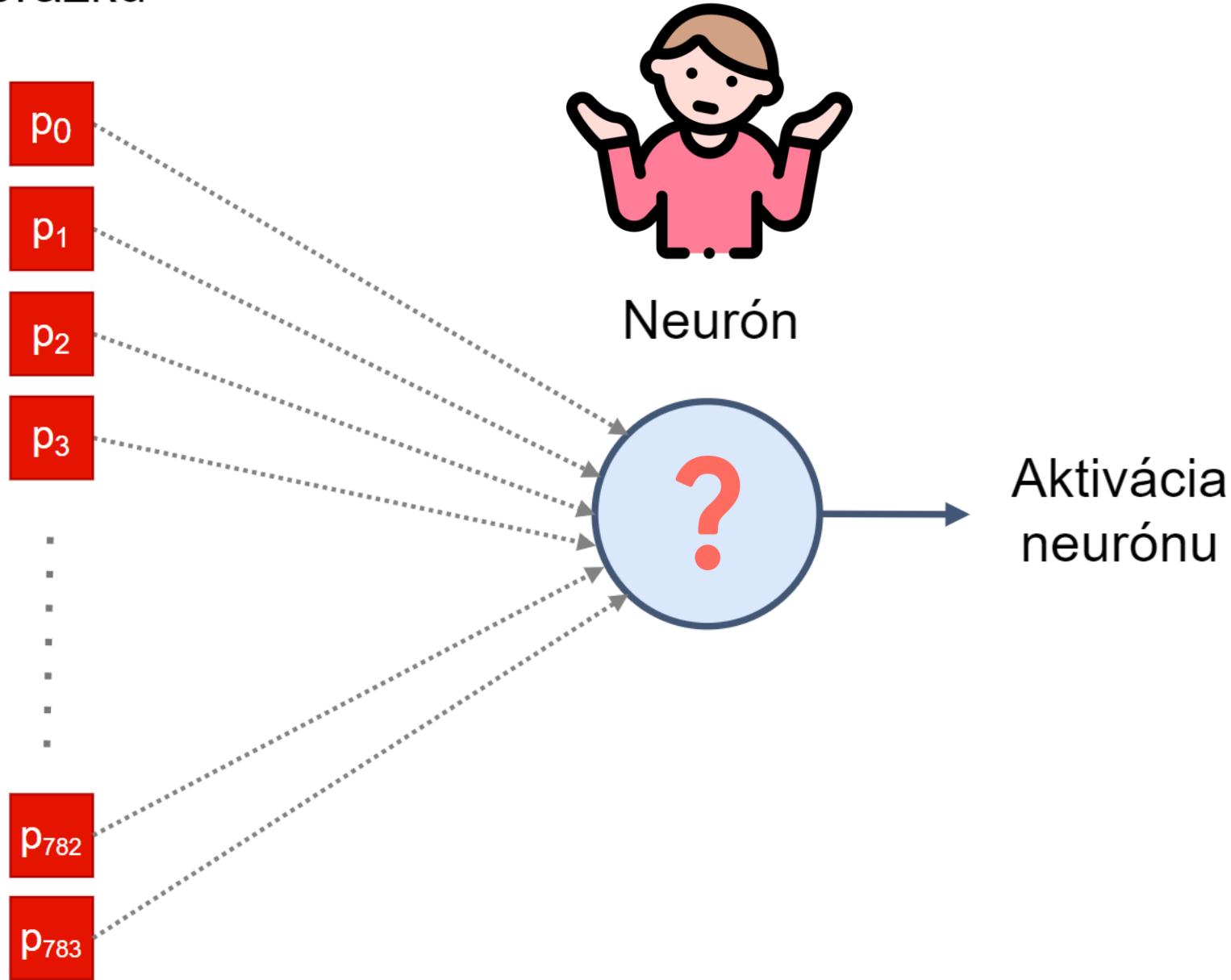








Pixely  
obrázku



# Ako funguje neurón?

- Do každého neurónu vstupuje 785 hodnôt = 784 hodnôt pixelov + 1 bias hodnota.

# Ako funguje neurón?

- Do každého neurónu vstupuje 785 hodnôt = 784 hodnôt pixelov + 1 bias hodnota.
- Každý pixel vstupujúci do neurónu je vynásobený svojou váhou.

# Ako funguje neurón?

- Do každého neurónu vstupuje 785 hodnôt = 784 hodnôt pixelov + 1 bias hodnota.
- Každý pixel vstupujúci do neurónu je vynásobený svojou váhou.
- Bias hodnota má váhu 1.

# Ako funguje neurón?

- Do každého neurónu vstupuje 785 hodnôt = 784 hodnôt pixelov + 1 bias hodnota.
- Každý pixel vstupujúci do neurónu je vynásobený svojou váhou.
- Bias hodnota má váhu 1.
- Vypočíta sa vážený súčet všetkých pixelov + bias.

# Pixely obrázku

$p_0$

$p_1$

$p_2$

$p_3$

$p_{782}$

$p_{783}$

# Pixely obrázku

$p_0$

$p_1$

$p_2$

$p_3$

$p_{782}$

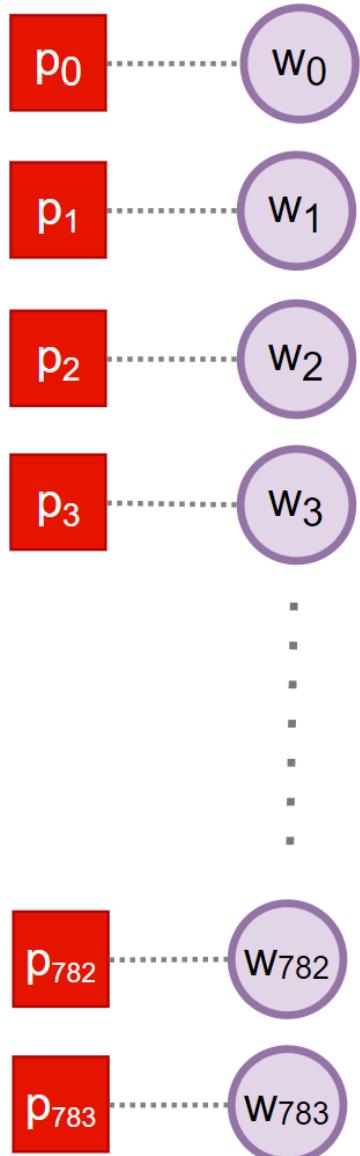
$p_{783}$

Neurón  $k$



Pixely  
obrázku

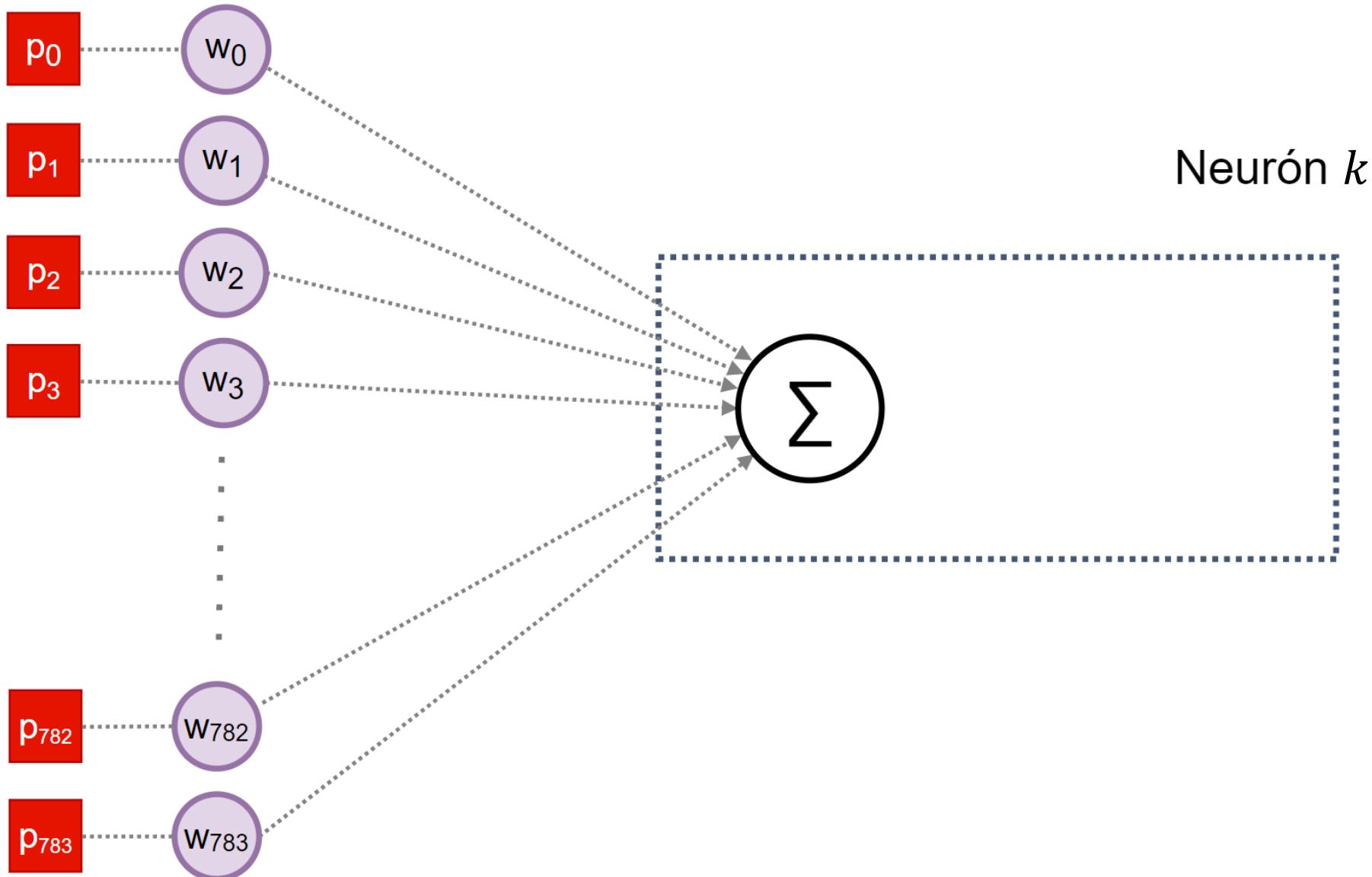
Váhy



Neurón  $k$

Pixely  
obrázku

Váhy

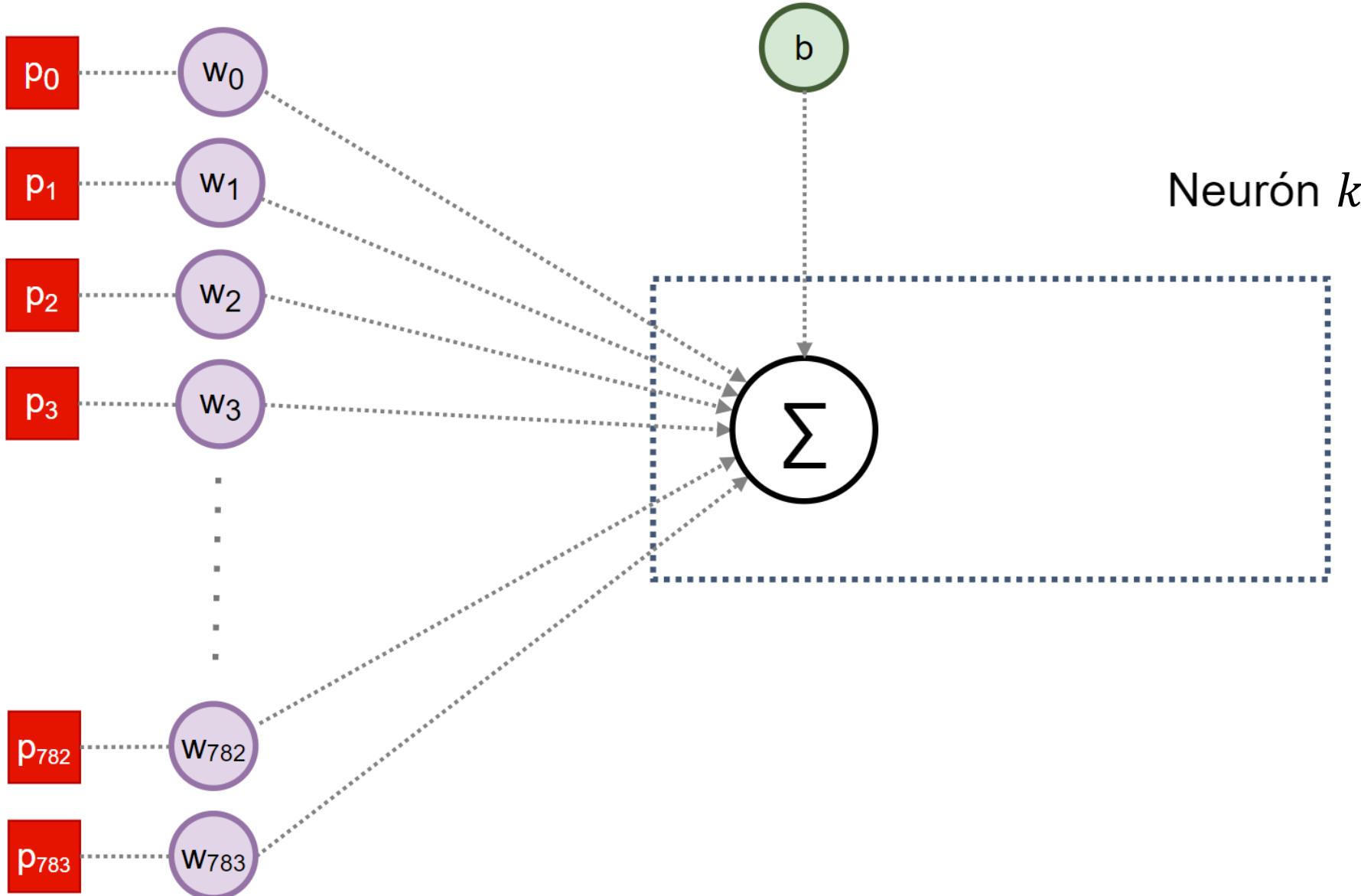


Neurón  $k$

Pixely  
obrázku

Váhy

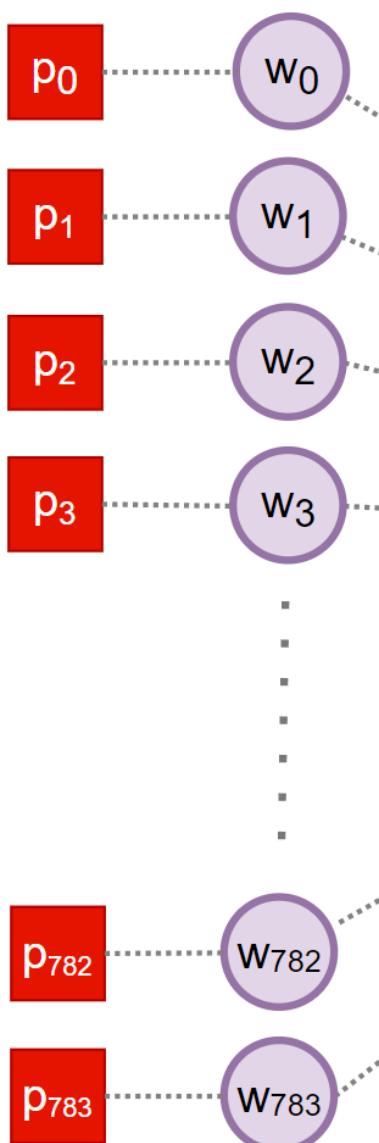
Bias pre neurón  $k$



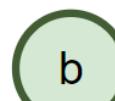
Neurón  $k$

Pixely  
obrázku

Váhy



Bias pre neurón  $k$

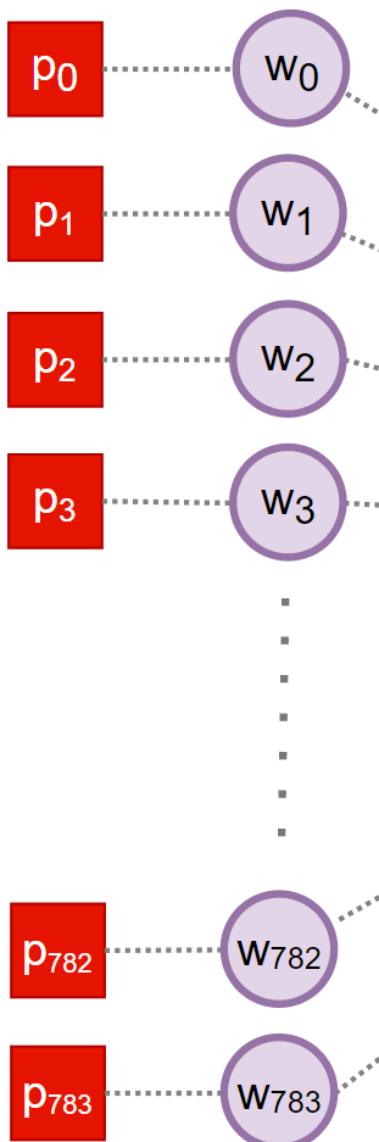


Neurón  $k$

$$S_k = \sum_{i=0}^{783} w_i p_i + b_k$$

Pixely  
obrázku

Váhy



Bias pre neurón  $k$



Neurón  $k$

$$S_k = \sum_{i=0}^{783} w_i p_i + b_k$$

Váha pixelu  $i$

Pixel  $i$

Bias pre  
neurón  $k$

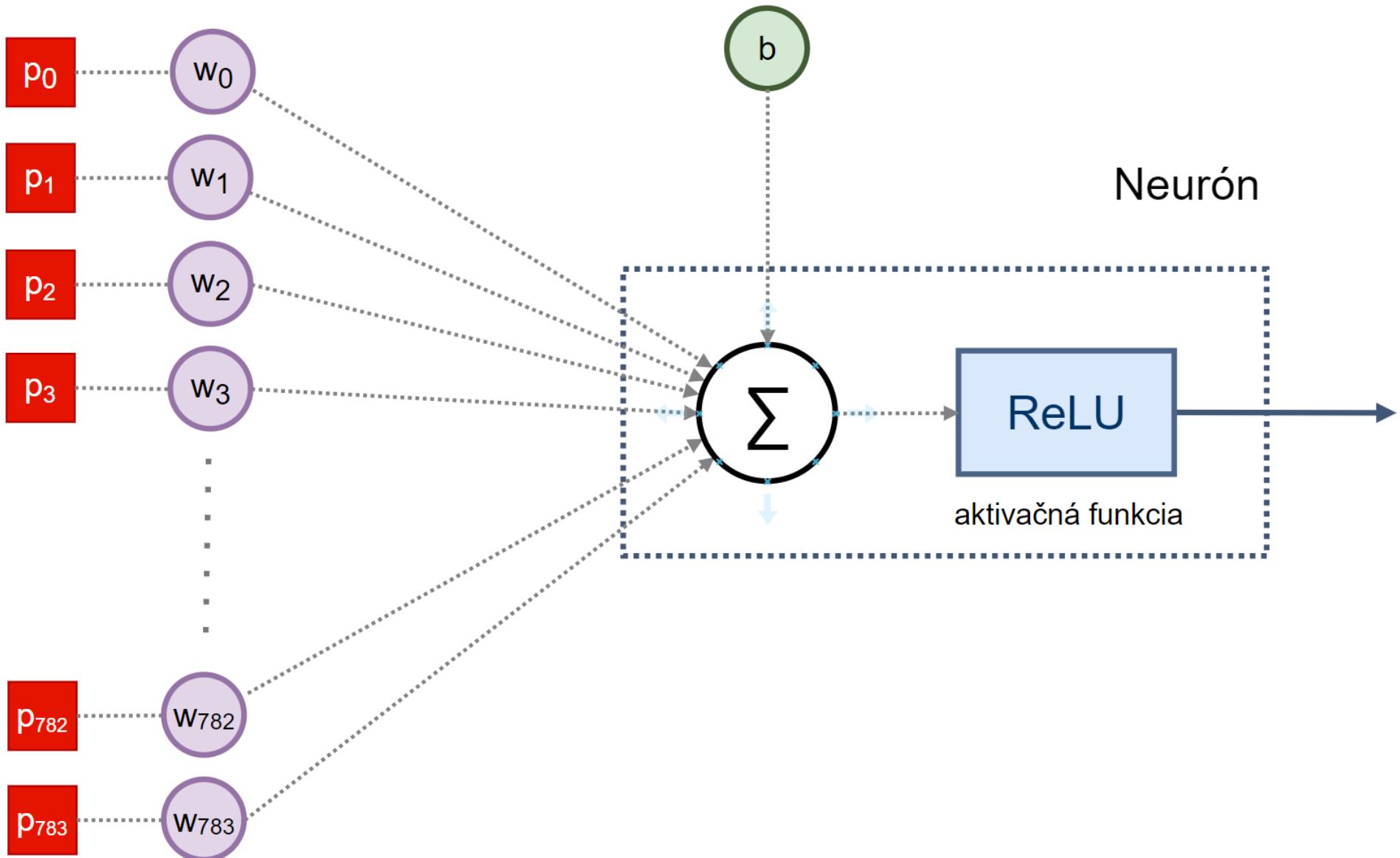


# Ako funguje neurón?

- Výsledok váženého súčtu sa pošle ako vstup do ReLU aktivačnej funkcie.

Pixely  
obrázku

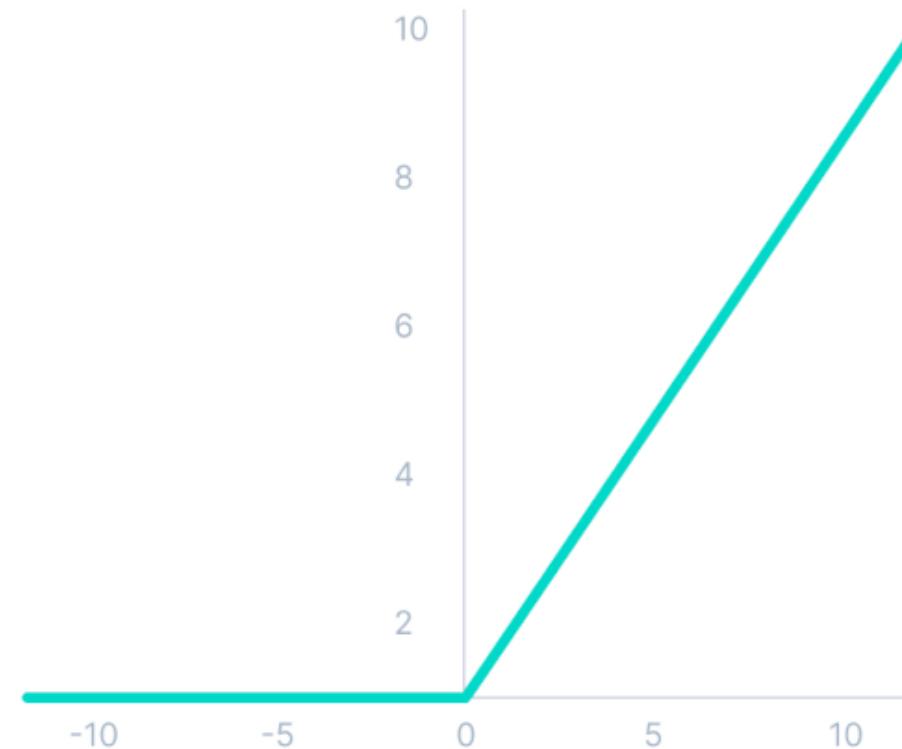
Váhy



# ReLU aktivačná funkcia

Funkcia je definovaná vzťahom:

$$f(x) = \begin{cases} 0 & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases}$$



# Váhy a bias hodnoty?

- Váhy a bias hodnoty sú uložené v 1D poliach v dodanom zdrojovom súbore *data.c*.
- **Počet váh:** 784 × 10
- **Počet bias hodnôt:** 10

# Váhy a bias hodnoty v súbore *data.c*

```
// pole bias hodnot (kazdy neuron ma 1 bias hodnotu)
double bias[NUMBER_OF_NEURONS] = { [0]: 0.63, [1]: 0.43, [2]: 0.53, [3]: 0.73, [4]: 0. }
```

# Váhy a bias hodnoty v súbore *data.c*

```
// pole bias hodnot (kazdy neuron ma 1 bias hodnotu)
double bias[NUMBER_OF_NEURONS] = { [0]: 0.63, [1]: 0.43, [2]: 0.53, [3]: 0.73, [4]: 0.
```



Kapacita pola 10

# Váhy a bias hodnoty v súbore *data.c*

```
// pole bias hodnot (kazdy neuron ma 1 bias hodnotu)
double bias[NUMBER_OF_NEURONS] = { [0]: 0.63, [1]: 0.43, [2]: 0.53, [3]: 0.73, [4]: 0.25, [5]: 0.35, [6]: 0.15, [7]: 0.2, [8]: 0.1, [9]: 0.3 };

// pole vah (spolu mame 10 neuronov, pocet vah 1 neuronu je IMAGE_SIZE=784)
double weights[NUMBER_OF_WEIGHTS] = {
    [0]: 0.39, [1]: 0.78, [2]: 0.80, [3]: 0.91, [4]: 0.20, [5]: 0.34, [6]: 0.77, [7]: 0.28, [8]: 0.15, [9]: 0.1, [10]: 0.1, [11]: 0.1, [12]: 0.1, [13]: 0.1, [14]: 0.1, [15]: 0.64, [16]: 0.72, [17]: 0.14, [18]: 0.61, [19]: 0.02, [20]: 0.24, [21]: 0.14, [22]: 0.1, [23]: 0.1, [24]: 0.1, [25]: 0.1, [26]: 0.1, [27]: 0.1, [28]: 0.1, [29]: 0.1, [30]: 0.84, [31]: 0.61, [32]: 0.30, [33]: 0.64, [34]: 0.52, [35]: 0.49, [36]: 0.97, [37]: 0.1, [38]: 0.1, [39]: 0.1, [40]: 0.1, [41]: 0.1, [42]: 0.1, [43]: 0.1, [44]: 0.1, [45]: 0.81, [46]: 0.92, [47]: 0.07, [48]: 0.95, [49]: 0.53, [50]: 0.09, [51]: 0.19, [52]: 0.1, [53]: 0.1, [54]: 0.1, [55]: 0.1, [56]: 0.1, [57]: 0.1, [58]: 0.1, [59]: 0.1, [60]: 0.97, [61]: 0.90, [62]: 0.85, [63]: 0.27, [64]: 0.54, [65]: 0.37, [66]: 0.75, [67]: 0.1, [68]: 0.1, [69]: 0.1, [70]: 0.1, [71]: 0.1, [72]: 0.1, [73]: 0.1, [74]: 0.1, [75]: 0.27, [76]: 0.73, [77]: 0.63, [78]: 0.35, [79]: 0.69, [80]: 0.17, [81]: 0.44, [82]: 0.1, [83]: 0.1, [84]: 0.1, [85]: 0.1, [86]: 0.1, [87]: 0.1, [88]: 0.1, [89]: 0.1, [90]: 0.59, [91]: 0.65, [92]: 0.85, [93]: 0.41, [94]: 0.88, [95]: 0.34, [96]: 0.74, [97]: 0.1, [98]: 0.1, [99]: 0.1 }
```

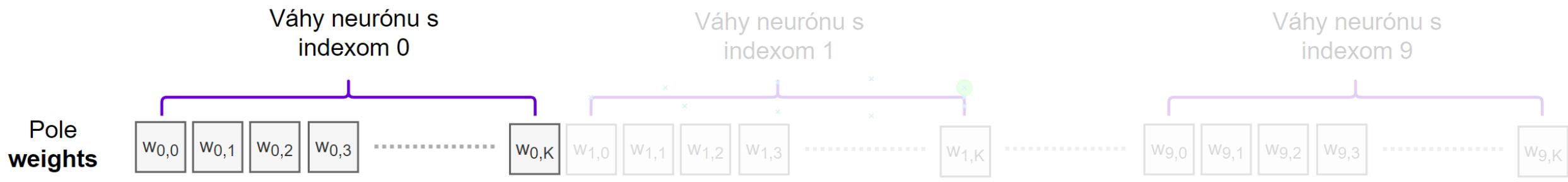
# Váhy a bias hodnoty v súbore *data.c*

```
// pole bias hodnot (kazdy neuron ma 1 bias hodnotu)
double bias[NUMBER_OF_NEURONS] = { [0]: 0.63, [1]: 0.43, [2]: 0.53, [3]: 0.73, [4]: 0.
```

// pole vah (spolu mame 10 neuronov, pocet vah 1 neuronu je IMAGE\_SIZE=784)

```
double weights[NUMBER_OF_WEIGHTS] = {  
    [0]: 0.39, [1]: 0.78, [2]: 0.80, [3]: 0.91, [4]: 0.20, [5]: 0.34, [6]: 0.77, [7]: 0.28,  
    [15]: 0.64, [16]: 0.72, [17]: 0.61, [18]: 0.61, [19]: 0.02, [20]: 0.24, [21]: 0.14, [22]: 0.14,  
    [30]: 0.84, [31]: 0.84, [32]: 0.84, [33]: 0.84, [34]: 0.52, [35]: 0.49, [36]: 0.97, [37]: 0.97,  
    [45]: 0.81, [46]: 0.81, [47]: 0.81, [48]: 0.81, [49]: 0.53, [50]: 0.09, [51]: 0.19, [52]: 0.19,  
    [60]: 0.97, [61]: 0.90, [62]: 0.90, [63]: 0.90, [64]: 0.54, [65]: 0.37, [66]: 0.75, [67]: 0.75,  
    [75]: 0.27, [76]: 0.73, [77]: 0.63, [78]: 0.35, [79]: 0.69, [80]: 0.17, [81]: 0.44, [82]: 0.44,  
    [90]: 0.59, [91]: 0.65, [92]: 0.85, [93]: 0.41, [94]: 0.88, [95]: 0.34, [96]: 0.74, [97]: 0.74}
```

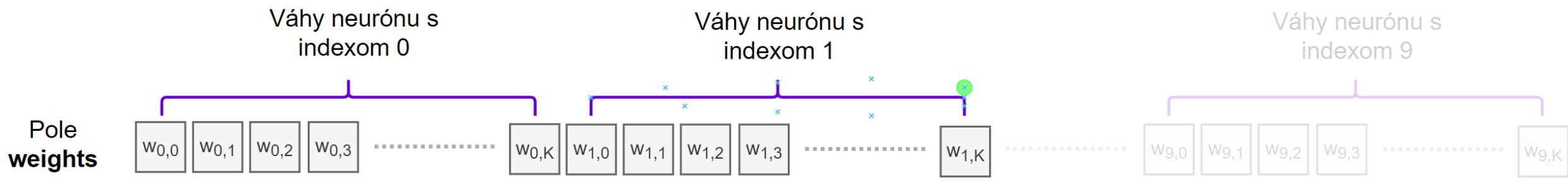
Kapacita pol'a je  $784 \times 10 =$   
7840 prvkov



### Legenda

$w_{i,j}$  : váha  $j$  neurónu  $i$

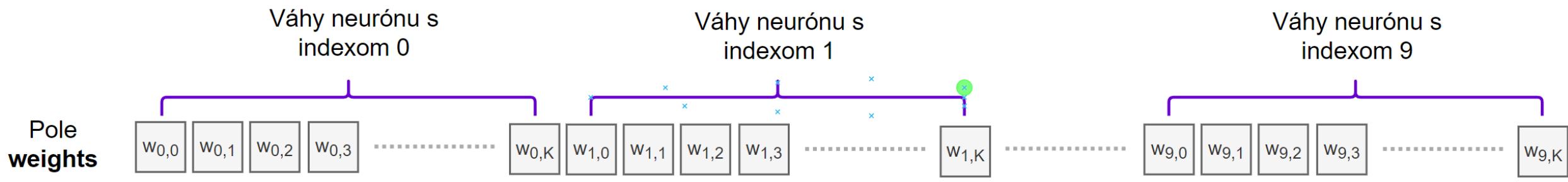
$K$  : počet pixelov obrázku



### Legenda

$w_{i,j}$  : váha  $j$  neurónu  $i$

$K$  : počet pixelov obrázku



### Legenda

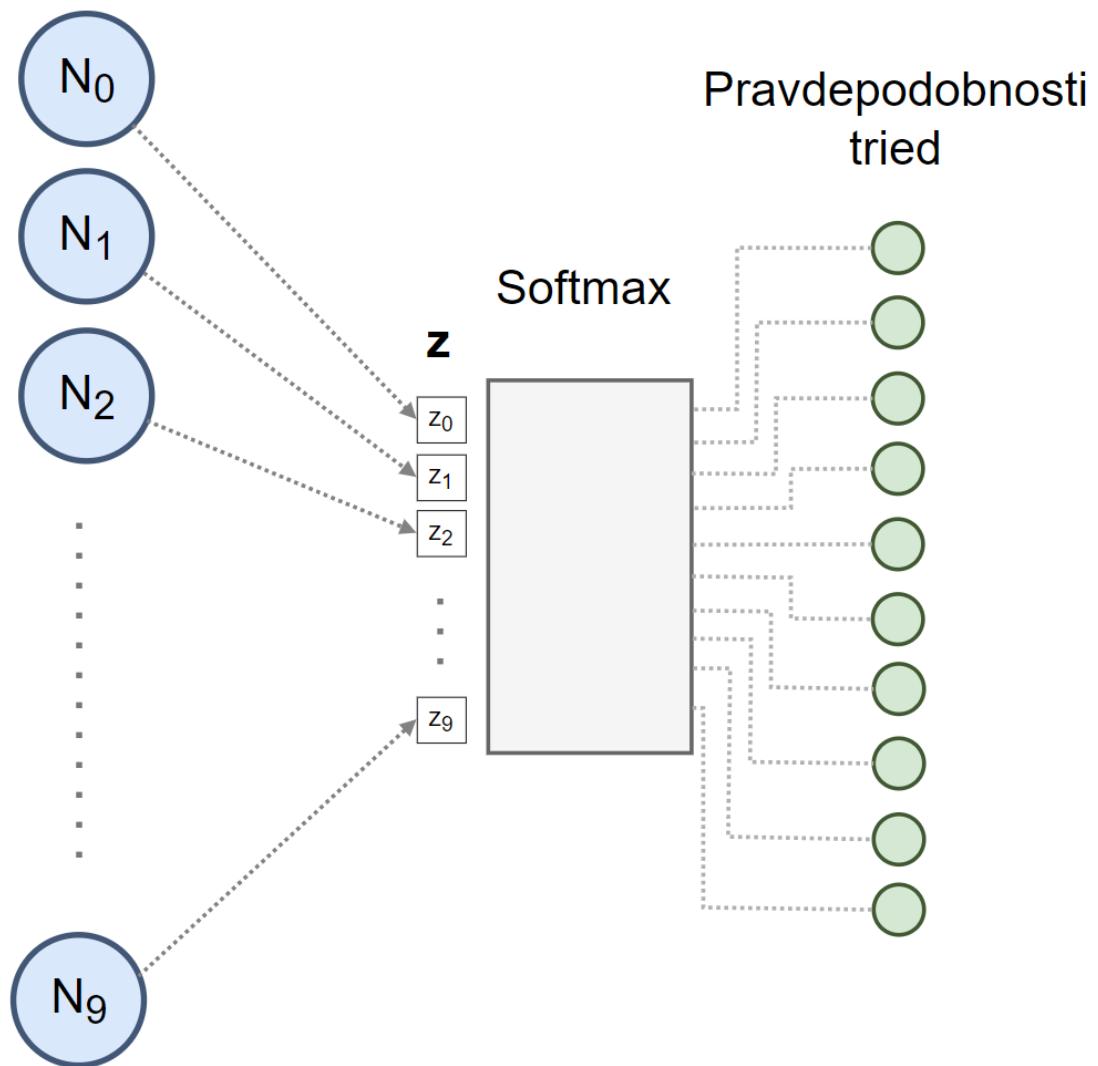
$w_{i,j}$  : váha  $j$  neurónu  $i$

$K$  : počet pixelov obrázku

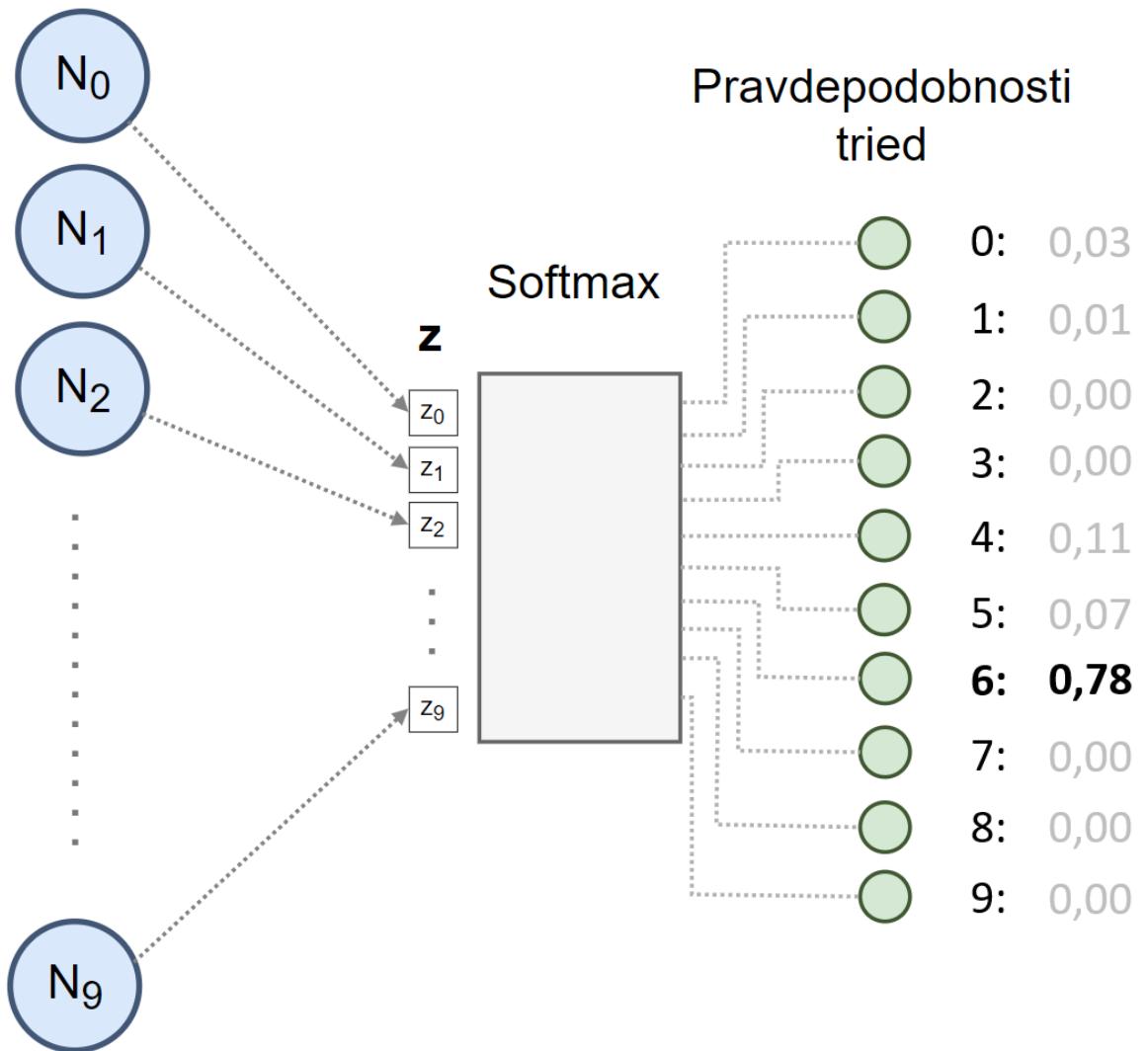
# Softmax funkcia

- Slúži na výpočet pravdepodobnosti klasifikácie pre každú triedu.
- Vstupom je pole 10 hodnôt (aktivácie všetkých neurónov).
- Výstupom je pole s pravdepodobnosťami (tiež 10 hodnôt).

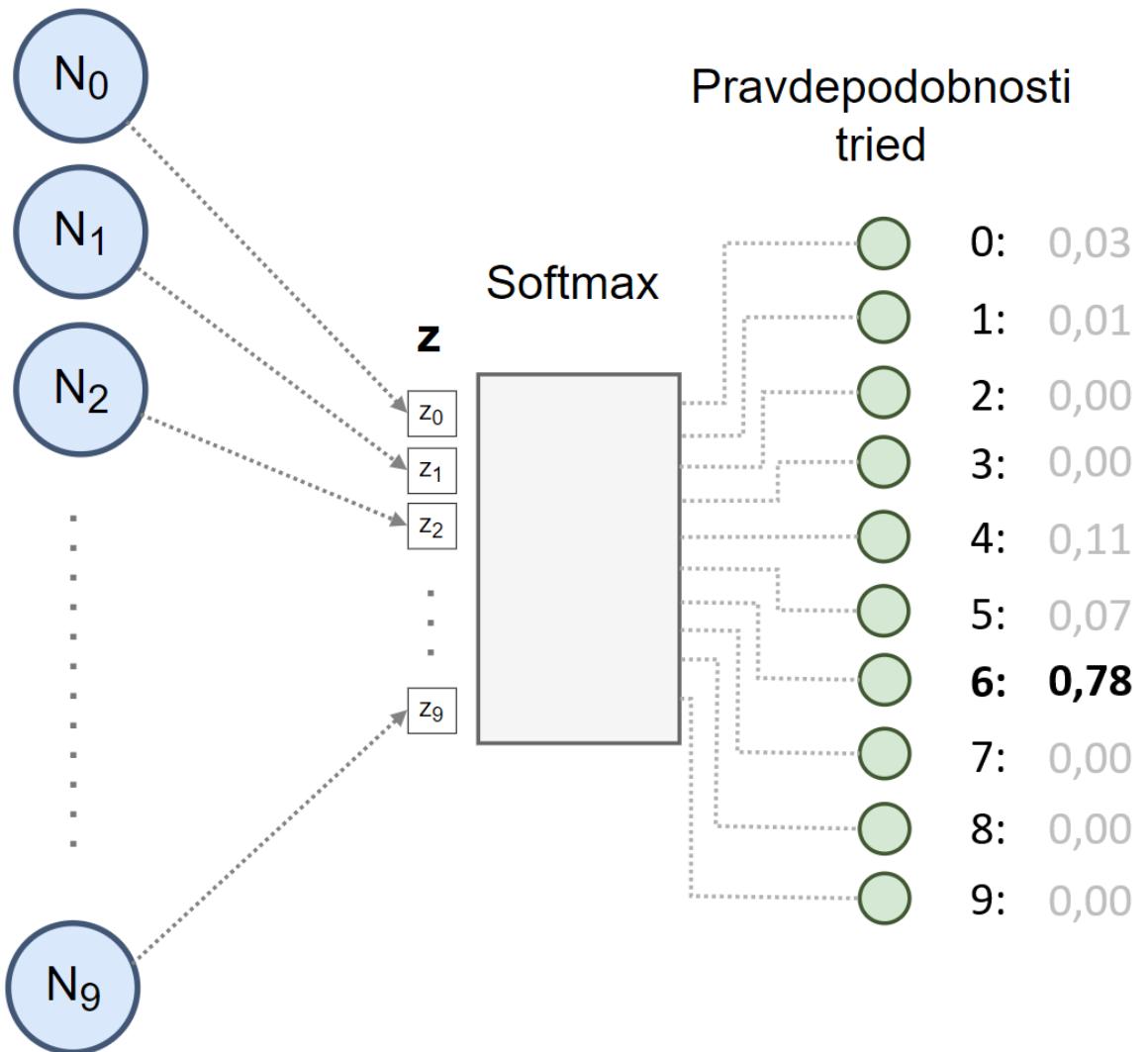
## Neuróny



# Neuróny



## Neuróny

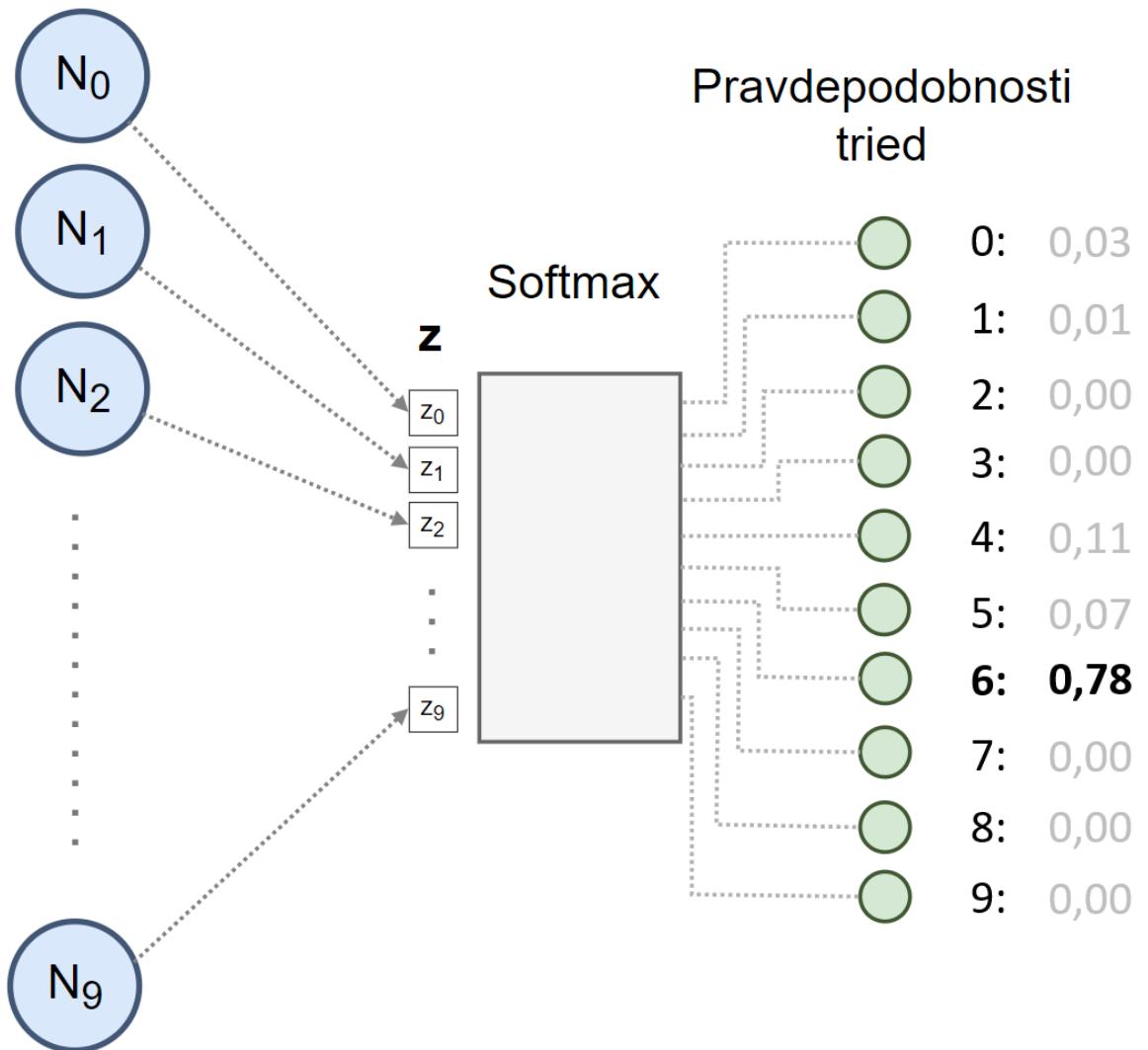


$$\text{softmax}(\mathbf{z})_i = \frac{e^{(z_i - \max(\mathbf{z}))}}{\sum_{j=0}^K e^{(z_j - \max(\mathbf{z}))}}$$

$$K = 9$$

$\mathbf{z}$  je vstupné pole do funkcie softmax()

## Neuróny



Najväčší prvok poľa  $\mathbf{z}$ .

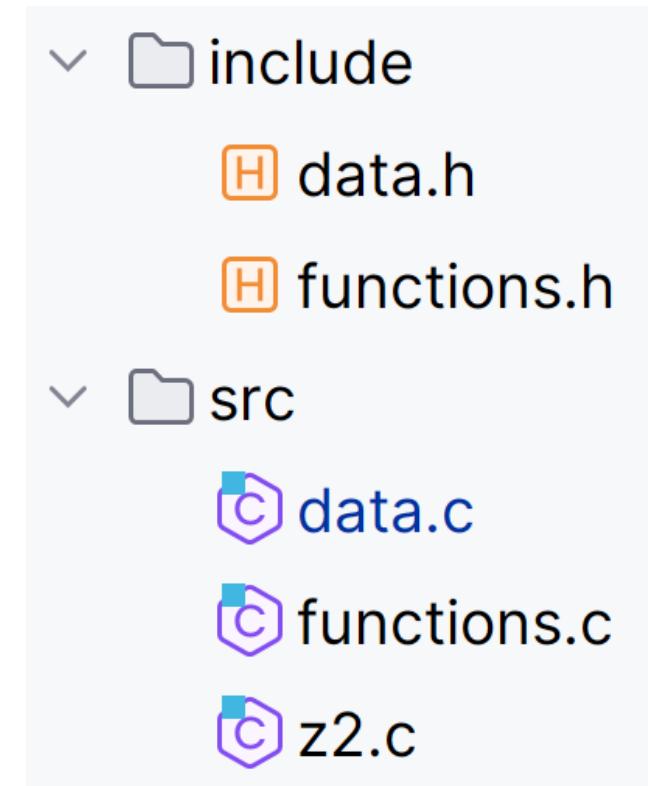
$$\text{softmax}(\mathbf{z})_i = \frac{e^{(z_i - \max(\mathbf{z}))}}{\sum_{j=0}^K e^{(z_j - \max(\mathbf{z}))}}$$

$$K = 9$$

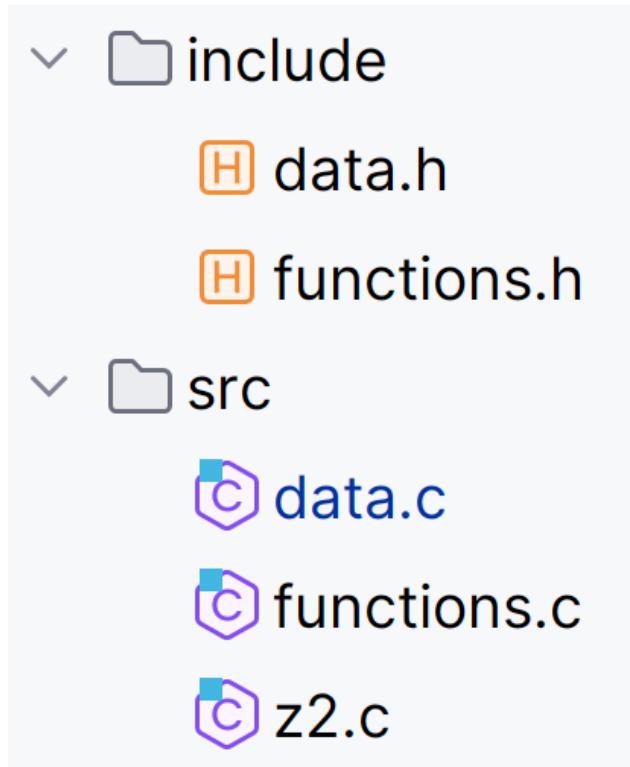
$\mathbf{z}$  je vstupné pole do funkcie softmax()

# Projekt

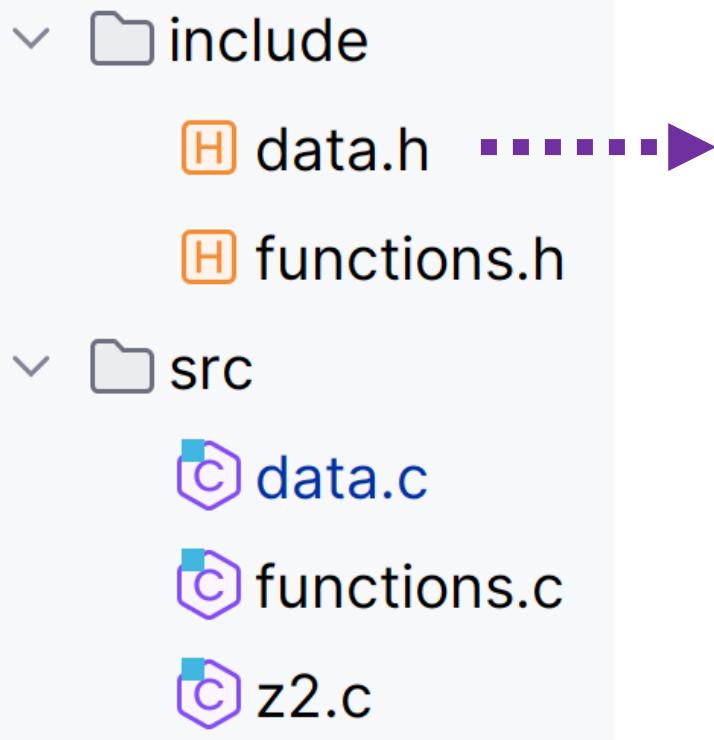
- Projekt sa skladá 5 súborov.
- Hlavičkové súbory:
  - data.h dodané
  - functions.h dodané
- Zdrojové súbory:
  - data.c dodané
  - functions.c dodané
  - z2.c



# Projekt



# Projekt



```
/*
 *  Makra, vahy a bias hodnoty
 */

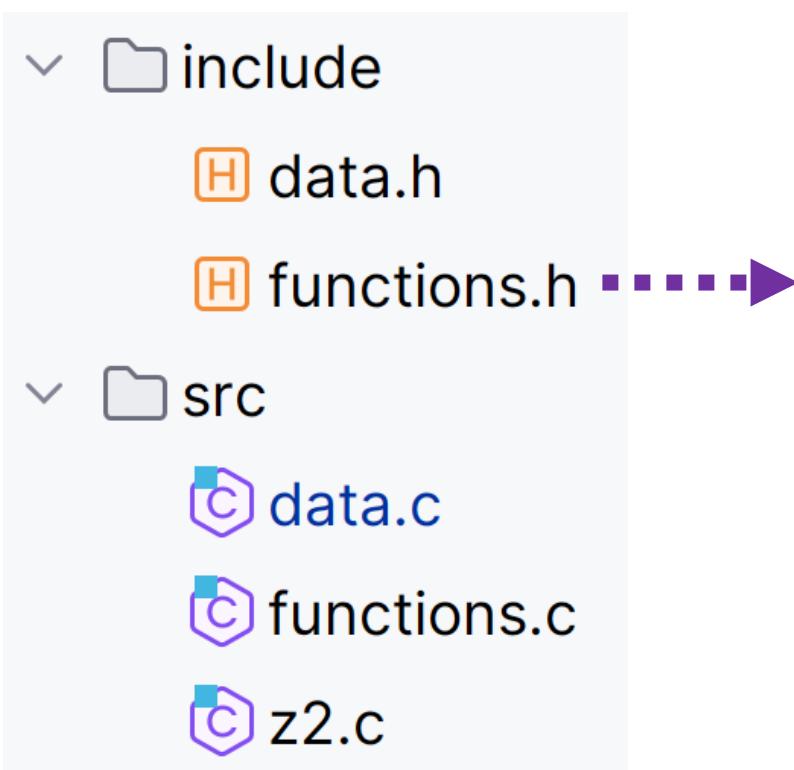
#ifndef DATA_H
#define DATA_H

#define IMAGE_WIDTH 28
#define IMAGE_HEIGHT 28
#define IMAGE_SIZE IMAGE_WIDTH * IMAGE_HEIGHT // pocet prvkov vstupu
#define NUMBER_OF_NEURONS 10 // pocet neuronov
#define NUMBER_OF_WEIGHTS NUMBER_OF_NEURONS * IMAGE_SIZE // pocet vah

extern double bias[NUMBER_OF_NEURONS]; // pole bias hodnoty
extern double weights[NUMBER_OF_WEIGHTS]; // pole vah (pocet vah)

#endif //DATA_H
```

# Projekt



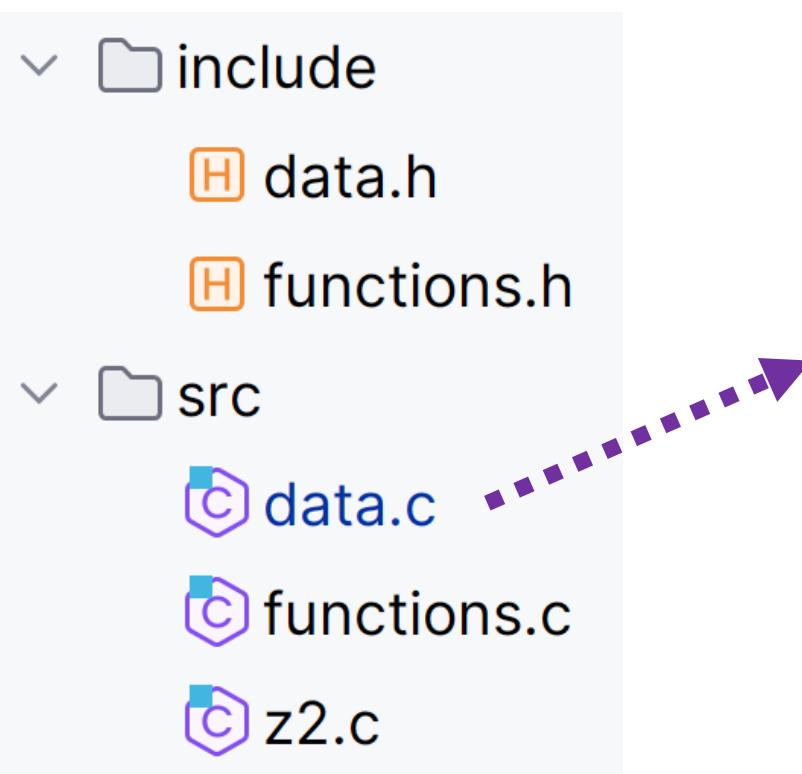
```
/*
 *  ReLU funkcia a pomocne funkcie
 */

#ifndef FUNCTIONS_H
#define FUNCTIONS_H

double relu( double x); // aktivacna funkcia neuronovej siete
void load_data(double *data, unsigned int len); // funkcia na nahrani
void print_image(const double* img, unsigned int img_width, unsig

#endif //FUNCTIONS_H
```

# Projekt

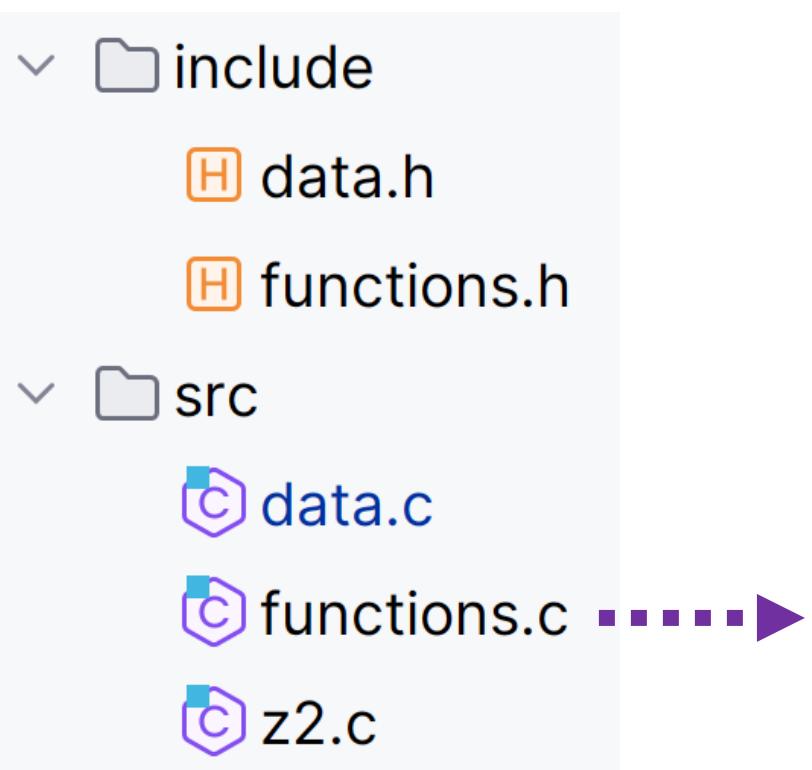


```
/*
 * Vahy a bias hodnoty
 */

#include "data.h"

// pole bias hodnot (kazdy neuron ma 1 bias hodnotu)
double bias[NUMBER_OF_NEURONS] = { [0]: 0.63, [1]: 0.43, [2]: 0.21, [3]: 0.55, [4]: 0.37, [5]: 0.72, [6]: 0.15, [7]: 0.89, [8]: 0.48, [9]: 0.32, [10]: 0.51, [11]: 0.29, [12]: 0.67, [13]: 0.42, [14]: 0.38, [15]: 0.64, [16]: 0.72, [17]: 0.14, [18]: 0.61, [19]: 0.01, [20]: 0.53, [21]: 0.25, [22]: 0.85, [23]: 0.36, [24]: 0.47, [25]: 0.22, [26]: 0.78, [27]: 0.1, [28]: 0.9, [29]: 0.44, [30]: 0.84, [31]: 0.61, [32]: 0.3, [33]: 0.64, [34]: 0.08, [35]: 0.5, [36]: 0.27, [37]: 0.82, [38]: 0.34, [39]: 0.49, [40]: 0.23, [41]: 0.75, [42]: 0.12, [43]: 0.93, [44]: 0.46, [45]: 0.81, [46]: 0.92, [47]: 0.07, [48]: 0.95, [49]: 0.02, [50]: 0.52, [51]: 0.28, [52]: 0.86, [53]: 0.31, [54]: 0.48, [55]: 0.24, [56]: 0.77, [57]: 0.13, [58]: 0.91, [59]: 0.45, [60]: 0.97, [61]: 0.9, [62]: 0.85, [63]: 0.27, [64]: 0.0, [65]: 0.54, [66]: 0.29, [67]: 0.87, [68]: 0.33, [69]: 0.49, [70]: 0.25, [71]: 0.79, [72]: 0.11, [73]: 0.94, [74]: 0.47, [75]: 0.27, [76]: 0.73, [77]: 0.63, [78]: 0.35, [79]: 0.0, [80]: 0.51, [81]: 0.26, [82]: 0.84, [83]: 0.32, [84]: 0.46, [85]: 0.22, [86]: 0.76, [87]: 0.14, [88]: 0.92, [89]: 0.48, [90]: 0.59, [91]: 0.65, [92]: 0.85, [93]: 0.41, [94]: 0.0, [95]: 0.56, [96]: 0.23, [97]: 0.82, [98]: 0.3, [99]: 0.44, [100]: 0.21, [101]: 0.74, [102]: 0.16, [103]: 0.9, [104]: 0.46, [105]: 0.53, [106]: 0.28, [107]: 0.81, [108]: 0.34, [109]: 0.47, [110]: 0.2, [111]: 0.72, [112]: 0.18, [113]: 0.89, [114]: 0.43, [115]: 0.58, [116]: 0.25, [117]: 0.83, [118]: 0.31, [119]: 0.45, [120]: 0.19, [121]: 0.71, [122]: 0.15, [123]: 0.87, [124]: 0.41, [125]: 0.55, [126]: 0.22, [127]: 0.8, [128]: 0.29, [129]: 0.43, [130]: 0.17, [131]: 0.68, [132]: 0.12, [133]: 0.84, [134]: 0.38, [135]: 0.52, [136]: 0.24, [137]: 0.76, [138]: 0.21, [139]: 0.88, [140]: 0.44, [141]: 0.57, [142]: 0.26, [143]: 0.79, [144]: 0.23, [145]: 0.86, [146]: 0.4, [147]: 0.54, [148]: 0.27, [149]: 0.71, [150]: 0.2, [151]: 0.83, [152]: 0.35, [153]: 0.48, [154]: 0.2, [155]: 0.69, [156]: 0.17, [157]: 0.81, [158]: 0.42, [159]: 0.56, [160]: 0.23, [161]: 0.77, [162]: 0.2, [163]: 0.85, [164]: 0.37, [165]: 0.49, [166]: 0.21, [167]: 0.73, [168]: 0.18, [169]: 0.82, [170]: 0.4, [171]: 0.53, [172]: 0.25, [173]: 0.75, [174]: 0.22, [175]: 0.84, [176]: 0.34, [177]: 0.46, [178]: 0.2, [179]: 0.67, [180]: 0.15, [181]: 0.8, [182]: 0.38, [183]: 0.5, [184]: 0.24, [185]: 0.72, [186]: 0.21, [187]: 0.83, [188]: 0.36, [189]: 0.47, [190]: 0.22, [191]: 0.69, [192]: 0.16, [193]: 0.79, [194]: 0.41, [195]: 0.52, [196]: 0.26, [197]: 0.74, [198]: 0.23, [199]: 0.81, [200]: 0.33, [201]: 0.49, [202]: 0.21, [203]: 0.66, [204]: 0.14, [205]: 0.77, [206]: 0.2, [207]: 0.82, [208]: 0.37, [209]: 0.51, [210]: 0.25, [211]: 0.71, [212]: 0.22, [213]: 0.84, [214]: 0.35, [215]: 0.53, [216]: 0.27, [217]: 0.73, [218]: 0.24, [219]: 0.8, [220]: 0.32, [221]: 0.48, [222]: 0.2, [223]: 0.64, [224]: 0.13, [225]: 0.76, [226]: 0.21, [227]: 0.83, [228]: 0.38, [229]: 0.55, [230]: 0.28, [231]: 0.7, [232]: 0.25, [233]: 0.81, [234]: 0.36, [235]: 0.52, [236]: 0.26, [237]: 0.68, [238]: 0.19, [239]: 0.75, [240]: 0.34, [241]: 0.57, [242]: 0.23, [243]: 0.72, [244]: 0.2, [245]: 0.8, [246]: 0.33, [247]: 0.54, [248]: 0.21, [249]: 0.65, [250]: 0.17, [251]: 0.78, [252]: 0.35, [253]: 0.59, [254]: 0.28, [255]: 0.7, [256]: 0.25, [257]: 0.82, [258]: 0.37, [259]: 0.56, [260]: 0.29, [261]: 0.67, [262]: 0.18, [263]: 0.74, [264]: 0.32, [265]: 0.58, [266]: 0.27, [267]: 0.71, [268]: 0.24, [269]: 0.81, [270]: 0.34, [271]: 0.55, [272]: 0.26, [273]: 0.69, [274]: 0.16, [275]: 0.76, [276]: 0.31, [277]: 0.57, [278]: 0.29, [279]: 0.73, [280]: 0.26, [281]: 0.8, [282]: 0.36, [283]: 0.53, [284]: 0.3, [285]: 0.66, [286]: 0.19, [287]: 0.74, [288]: 0.33, [289]: 0.59, [290]: 0.27, [291]: 0.71, [292]: 0.24, [293]: 0.81, [294]: 0.35, [295]: 0.56, [296]: 0.29, [297]: 0.68, [298]: 0.17, [299]: 0.75, [2000]: 0.38, [2001]: 0.62, [2002]: 0.45, [2003]: 0.51, [2004]: 0.2, [2005]: 0.7, [2006]: 0.34, [2007]: 0.58, [2008]: 0.26, [2009]: 0.64, [2010]: 0.18, [2011]: 0.72, [2012]: 0.31, [2013]: 0.55, [2014]: 0.28, [2015]: 0.67, [2016]: 0.15, [2017]: 0.79, [2018]: 0.36, [2019]: 0.52, [2020]: 0.25, [2021]: 0.69, [2022]: 0.12, [2023]: 0.77, [2024]: 0.33, [2025]: 0.57, [2026]: 0.27, [2027]: 0.7, [2028]: 0.14, [2029]: 0.8, [2030]: 0.38, [2031]: 0.54, [2032]: 0.29, [2033]: 0.65, [2034]: 0.17, [2035]: 0.73, [2036]: 0.35, [2037]: 0.56, [2038]: 0.28, [2039]: 0.68, [2040]: 0.13, [2041]: 0.76, [2042]: 0.32, [2043]: 0.58, [2044]: 0.26, [2045]: 0.72, [2046]: 0.16, [2047]: 0.81, [2048]: 0.37, [2049]: 0.53, [2050]: 0.24, [2051]: 0.66, [2052]: 0.11, [2053]: 0.78, [2054]: 0.34, [2055]: 0.59, [2056]: 0.23, [2057]: 0.74, [2058]: 0.18, [2059]: 0.8, [2060]: 0.39, [2061]: 0.51, [2062]: 0.25, [2063]: 0.69, [2064]: 0.1, [2065]: 0.76, [2066]: 0.31, [2067]: 0.57, [2068]: 0.22, [2069]: 0.71, [2070]: 0.15, [2071]: 0.82, [2072]: 0.35, [2073]: 0.55, [2074]: 0.21, [2075]: 0.67, [2076]: 0.08, [2077]: 0.79, [2078]: 0.32, [2079]: 0.58, [2080]: 0.2, [2081]: 0.73, [2082]: 0.12, [2083]: 0.8, [2084]: 0.36, [2085]: 0.52, [2086]: 0.19, [2087]: 0.75, [2088]: 0.3, [2089]: 0.57, [2090]: 0.23, [2091]: 0.7, [2092]: 0.07, [2093]: 0.81, [2094]: 0.34, [2095]: 0.54, [2096]: 0.2, [2097]: 0.68, [2098]: 0.05, [2099]: 0.77, [2100]: 0.31, [2101]: 0.59, [2102]: 0.18, [2103]: 0.72, [2104]: 0.09, [2105]: 0.8, [2106]: 0.37, [2107]: 0.56, [2108]: 0.22, [2109]: 0.74, [2110]: 0.06, [2111]: 0.82, [2112]: 0.33, [2113]: 0.58, [2114]: 0.17, [2115]: 0.71, [2116]: 0.08, [2117]: 0.8, [2118]: 0.38, [2119]: 0.55, [2120]: 0.21, [2121]: 0.69, [2122]: 0.04, [2123]: 0.76, [2124]: 0.3, [2125]: 0.59, [2126]: 0.16, [2127]: 0.73, [2128]: 0.07, [2129]: 0.8, [2130]: 0.39, [2131]: 0.53, [2132]: 0.2, [2133]: 0.7, [2134]: 0.03, [2135]: 0.81, [2136]: 0.35, [2137]: 0.57, [2138]: 0.19, [2139]: 0.72, [2140]: 0.05, [2141]: 0.8, [2142]: 0.4, [2143]: 0.55, [2144]: 0.25, [2145]: 0.71, [2146]: 0.02, [2147]: 0.8, [2148]: 0.36, [2149]: 0.58, [2150]: 0.2, [2151]: 0.73, [2152]: 0.01, [2153]: 0.8, [2154]: 0.41, [2155]: 0.56, [2156]: 0.24, [2157]: 0.7, [2158]: 0.0, [2159]: 0.8, [2160]: 0.45, [2161]: 0.54, [2162]: 0.23, [2163]: 0.72, [2164]: 0.0, [2165]: 0.8, [2166]: 0.42, [2167]: 0.57, [2168]: 0.21, [2169]: 0.71, [2170]: 0.0, [2171]: 0.8, [2172]: 0.44, [2173]: 0.55, [2174]: 0.22, [2175]: 0.7, [2176]: 0.0, [2177]: 0.8, [2178]: 0.43, [2179]: 0.56, [2180]: 0.2, [2181]: 0.73, [2182]: 0.0, [2183]: 0.8, [2184]: 0.46, [2185]: 0.54, [2186]: 0.21, [2187]: 0.71, [2188]: 0.0, [2189]: 0.8, [2190]: 0.47, [2191]: 0.53, [2192]: 0.2, [2193]: 0.7, [2194]: 0.0, [2195]: 0.8, [2196]: 0.48, [2197]: 0.52, [2198]: 0.2, [2199]: 0.72, [2200]: 0.0, [2201]: 0.8, [2202]: 0.49, [2203]: 0.51, [2204]: 0.2, [2205]: 0.7, [2206]: 0.0, [2207]: 0.8, [2208]: 0.5, [2209]: 0.5, [2210]: 0.2, [2211]: 0.73, [2212]: 0.0, [2213]: 0.8, [2214]: 0.51, [2215]: 0.5, [2216]: 0.2, [2217]: 0.71, [2218]: 0.0, [2219]: 0.8, [2220]: 0.52, [2221]: 0.5, [2222]: 0.2, [2223]: 0.7, [2224]: 0.0, [2225]: 0.8, [2226]: 0.53, [2227]: 0.5, [2228]: 0.2, [2229]: 0.72, [2230]: 0.0, [2231]: 0.8, [2232]: 0.54, [2233]: 0.5, [2234]: 0.2, [2235]: 0.71, [2236]: 0.0, [2237]: 0.8, [2238]: 0.55, [2239]: 0.5, [2240]: 0.2, [2241]: 0.7, [2242]: 0.0, [2243]: 0.8, [2244]: 0.56, [2245]: 0.5, [2246]: 0.2, [2247]: 0.73, [2248]: 0.0, [2249]: 0.8, [2250]: 0.57, [2251]: 0.5, [2252]: 0.2, [2253]: 0.71, [2254]: 0.0, [2255]: 0.8, [2256]: 0.58, [2257]: 0.5, [2258]: 0.2, [2259]: 0.7, [2260]: 0.0, [2261]: 0.8, [2262]: 0.59, [2263]: 0.5, [2264]: 0.2, [2265]: 0.72, [2266]: 0.0, [2267]: 0.8, [2268]: 0.6, [2269]: 0.5, [2270]: 0.2, [2271]: 0.7, [2272]: 0.0, [2273]: 0.8, [2274]: 0.61, [2275]: 0.5, [2276]: 0.2, [2277]: 0.73, [2278]: 0.0, [2279]: 0.8, [2280]: 0.62, [2281]: 0.5, [2282]: 0.2, [2283]: 0.71, [2284]: 0.0, [2285]: 0.8, [2286]: 0.63, [2287]: 0.5, [2288]: 0.2, [2289]: 0.7, [2290]: 0.0, [2291]: 0.8, [2292]: 0.64, [2293]: 0.5, [2294]: 0.2, [2295]: 0.72, [2296]: 0.0, [2297]: 0.8, [2298]: 0.65, [2299]: 0.5, [2200]: 0.2, [2201]: 0.7, [2202]: 0.0, [2203]: 0.8, [2204]: 0.66, [2205]: 0.5, [2206]: 0.2, [2207]: 0.73, [2208]: 0.0, [2209]: 0.8, [2210]: 0.67, [2211]: 0.5, [2212]: 0.2, [2213]: 0.71, [2214]: 0.0, [2215]: 0.8, [2216]: 0.68, [2217]: 0.5, [2218]: 0.2, [2219]: 0.7, [2220]: 0.0, [2221]: 0.8, [2222]: 0.69, [2223]: 0.5, [2224]: 0.2, [2225]: 0.72, [2226]: 0.0, [2227]: 0.8, [2228]: 0.7, [2229]: 0.5, [2230]: 0.2, [2231]: 0.7, [2232]: 0.0, [2233]: 0.8, [2234]: 0.71, [2235]: 0.5, [2236]: 0.2, [2237]: 0.73, [2238]: 0.0, [2239]: 0.8, [2240]: 0.72, [2241]: 0.5, [2242]: 0.2, [2243]: 0.71, [2244]: 0.0, [2245]: 0.8, [2246]: 0.73, [2247]: 0.5, [2248]: 0.2, [2249]: 0.7, [2250]: 0.0, [2251]: 0.8, [2252]: 0.74, [2253]: 0.5, [2254]: 0.2, [2255]: 0.72, [2256]: 0.0, [2257]: 0.8, [2258]: 0.75, [2259]: 0.5, [2260]: 0.2, [2261]: 0.7, [2262]: 0.0, [2263]: 0.8, [2264]: 0.76, [2265]: 0.5, [2266]: 0.2, [2267]: 0.73, [2268]: 0.0, [2269]: 0.8, [2270]: 0.77, [2271]: 0.5, [2272]: 0.2, [2273]: 0.71, [2274]: 0.0, [2275]: 0.8, [2276]: 0.78, [2277]: 0.5, [2278]: 0.2, [2279]: 0.7, [2280]: 0.0, [2281]: 0.8, [2282]: 0.79, [2283]: 0.5, [2284]: 0.2, [2285]: 0.72, [2286]: 0.0, [2287]: 0.8, [2288]: 0.8, [2289]: 0.5, [2290]: 0.2, [2291]: 0.7, [2292]: 0.0, [2293]: 0.8, [2294]: 0.81, [2295]: 0.5, [2296]: 0.2, [2297]: 0.73, [2298]: 0.0, [2299]: 0.8, [2300]: 0.82, [2301]: 0.5, [2302]: 0.2, [2303]: 0.71, [2304]: 0.0, [2305]: 0.8, [2306]: 0.83, [2307]: 0.5, [2308]: 0.2, [2309]: 0.7, [2310]: 0.0, [2311]: 0.8, [2312]: 0.84, [2313]: 0.5, [2314]: 0.2, [2315]: 0.72, [2316]: 0.0, [2317]: 0.8, [2318]: 0.85, [2319]: 0.5, [2320]: 0.2, [2321]: 0.7, [2322]: 0.0, [2323]: 0.8, [2324]: 0.86, [2325]: 0.5, [2326]: 0.2, [2327]: 0.73, [2328]: 0.0, [2329]: 0.8, [2330]: 0.87, [2331]: 0.5, [2332]: 0.2, [2333]: 0.71, [2334]: 0.0, [2335]: 0.8, [2336]: 0.88, [2337]: 0.5, [2338]: 0.2, [2339]: 0.7, [2340]: 0.0, [2341]: 0.8, [2342]: 0.89, [2343]: 0.5, [2344]: 0.2, [2345]: 0.72, [2346]: 0.0, [2347]: 0.8, [2348]: 0.9, [2349]: 0.5, [2350]: 0.2, [2351]: 0.7, [2352]: 0.0, [2353]: 0.8, [2354]: 0.91, [2355]: 0.5, [2356]: 0.2, [2357]: 0.73, [2358]: 0.0, [2359]: 0.8, [2360]: 0.92, [2361]: 0.5, [2362]: 0.2, [2363]: 0.71, [2364]: 0.0, [2365]: 0.8, [2366]: 0.93, [2367]: 0.5, [2368]: 0.2, [2369]: 0.7, [2370]: 0.0, [2371]: 0.8, [2372]: 0.94, [2373]: 0.5, [2374]: 0.2, [2375]: 0.72, [2376]: 0.0, [2377]: 0.8, [2378]: 0.95, [2379]: 0.5, [2380]: 0.2, [2381]: 0.7, [2382]: 0.0, [2383]: 0.8, [2384]: 0.96, [2385]: 0.5, [2386]: 0.2, [2387]: 0.73, [2388]: 0.0, [2389]: 0.8, [2390]: 0.97, [2391]: 0.5, [2392]: 0.2, [2393]: 0.71, [2394]: 0.0, [2395]: 0.8, [2396]: 0.98, [2397]: 0.5, [2398]: 0.2, [2399]: 0.7, [2400]: 0.0, [2401]: 0.8, [2402]: 0.99, [2403]: 0.5, [2404]: 0.2, [2405]: 0.72, [2406]: 0.0, [2407]: 0.8, [2408]: 1.0, [2409]: 0.5, [2410]: 0.2, [2411]: 0.7, [2412]: 0.0, [2413]: 0.8, [2414]: 1.01, [2415]: 0.5, [2416]: 0.2, [2417]: 0.73, [2418]: 0.0, [2419]: 0.8, [2420]: 1.02, [2421]: 0.5, [2422]: 0.2, [2423]: 0.71, [2424]: 0.0, [2425]: 0.8, [2426]: 1.03, [2427]: 0.5, [2428]: 0.2, [2429]: 0.7, [2430]: 0.0, [2431]: 0.8, [2432]: 1.04, [2433]: 0.5, [2434]: 0.2, [2435]: 0.72, [2436]: 0.0, [2437]: 0.8, [2438]: 1.05, [2439]: 0.5, [2440]: 0.2, [2441]: 0.7, [2442]: 0.0, [2443]: 0.8, [2444]: 1.06, [2445]: 0.5, [2446]: 0.2, [2447]: 0.73, [2448]: 0.0, [2449]: 0.8, [2450]: 1.07, [2451]: 0.5, [2452]: 0.2, [2453]: 0.71, [2454]: 0.0, [2455]: 0.8, [2456]: 1.08, [2457]: 0.5, [2458]: 0.2, [2459]: 0.7, [2460]: 0.0, [2461]: 0.8, [2462]: 1.09, [2463]: 0.5, [2464]: 0.2, [2465]: 0.72, [2466]: 0.0, [2467]: 0.8, [2468]: 1.1, [2469]: 0.5, [2470]: 0.2, [2471]: 0.7, [2472]: 0.0, [2473]: 0.8, [2474]: 1.11, [2475]: 0.5, [2476]: 0.2, [2477]: 0.73, [2478]: 0.0, [2479]: 0.8, [2480]: 1.12, [2481]: 0.5, [2482]: 0.2, [2483]: 0.71, [2484]: 0.0, [2485]: 0.8, [2486]: 1.13, [2487]: 0.5, [2488]: 0.2, [2489]: 0.7, [2490]: 0.0, [2491]: 0.8, [2492]: 1.14, [2493]: 0.5, [2494]: 0.2, [2495]: 0.72, [2496]: 0.0, [2497]: 0.8, [2498]: 1.15, [2499]: 0.5, [2500]: 0.2, [2501]: 0.7, [2502]: 0.0, [2503]: 0.8, [2504]: 1.16, [2505]: 0.5, [2506]: 0.2, [2507]: 0.73, [2508]: 0.0, [2509]: 0.8, [2510]: 1.17, [2511]: 0.5, [2512]: 0.2, [2513]: 0.71, [2514]: 0.0, [2515]: 0.8, [2516]: 1.18, [2517]: 0.5, [2518]: 0.2, [2519]: 0.7, [2520]: 0.0, [2521]: 0.8, [2522]: 1.19, [2523]: 0.5, [2524]: 0.2, [2525]: 0.72, [2526]: 0.0, [2527]: 0.8, [2528]: 1.2, [2529]: 0.5, [2530]: 0.2, [2531]: 0.7, [2532]: 0.0, [2533]: 0.8, [2534]: 1.21, [2535]: 0.5, [2536]: 0.2, [2537]: 0.73, [2538]: 0.0, [2539]: 0.8, [2540]: 1.22, [2541]: 0.5, [2542]: 0.2, [2543]: 0.71, [2544]: 0.0, [2545]: 0.8, [2546]: 1.23, [2547]: 0.5, [2548]: 0.2, [2549]: 0.7, [2550]: 0.0, [2551]: 0.8, [2552]: 1.24, [2553]: 0.5, [2554]: 0.2, [2555]: 0.72, [2556]: 0.0, [2557]: 0.8, [2558]: 1.25, [2559]: 0.5, [2560]: 0.2, [2561]: 0.7, [2562]: 0.0, [2563]: 0.8, [2564]: 1.26, [2565]: 0.5, [2566]: 0.2, [2567]: 0.73, [2568]: 0.0, [2569]: 0.8, [2570]: 1.27, [2571]: 0.5, [2572]: 0.2, [2573]:
```

# Projekt



```
/*
 *  ReLU funkcia a pomocne funkcie
 */

> #include ...

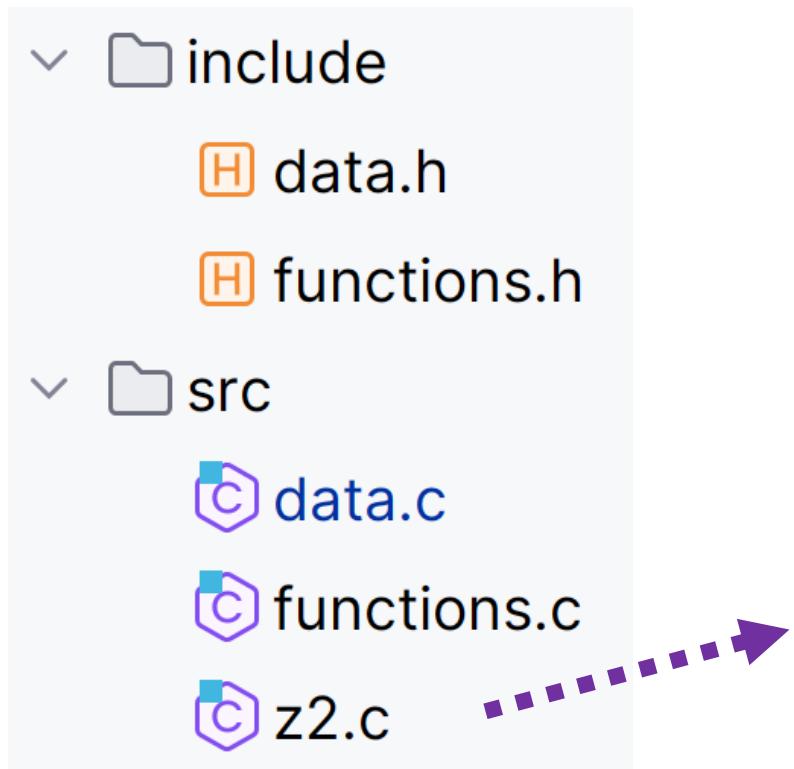
// ReLU funkcia
double relu(const double x){
    if(x>0){
        return x;
    }
    return 0;
}

// funkcia na nacitanie hodnot pola 'data' s dlzkou 'len' zo stanicu
void load_data(double *data, const unsigned int len){
    for(unsigned int i = 0; i< len;i++){
        scanf( format: "%lf", &data[i]);
    }
}

// funkcia na pomocny vypis obrazku (ocakavaju sa pixely v rozsahu 0-255)
void print_image(const double* img, unsigned int img_width, unsigned int img_height){
    char white[3] = "WW";
    char black[3] = "...";
    for(unsigned int i=0; i<img_height;i++){

```

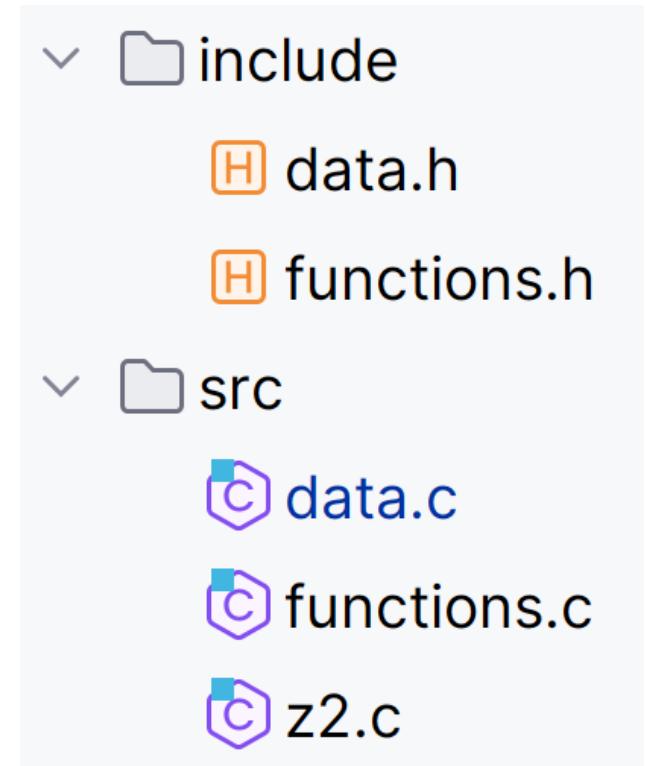
# Projekt



Tento súbor sa  
odovzdáva ako jediný.

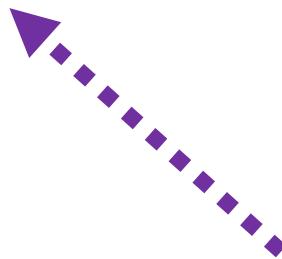
# CLion konfigurácia projektu

```
1 cmake_minimum_required(VERSION 3.27)
2 project(z2 C)
3
4 set(CMAKE_C_STANDARD 11)
5 set (CMAKE_C_FLAGS "${CMAKE_C_FLAGS} -Wall -Wextra -lm")
6
7 add_executable(z2 src/functions.c src/data.c src/z2.c)
8 target_include_directories(z2 PRIVATE ${PROJECT_SOURCE_DIR}/include)
```



# Správne vloženie hlavičkových súborov do súboru z2.c

```
#include "functions.h" // ReLU funkcia a pomocne funkcie  
#include "data.h" // Makra, vahy a bias hodnoty
```



Do zdrojového súboru **z2.c**, ktorý odovzdávate je nutné vložiť hlavičkové súbory **len pomocou názvu súboru bez cesty!!!**

# Správne vloženie hlavičkových súborov do súboru z2.c

```
#include "C:/prog2/functions.h" // ReLU funkcia a pomocne funkcie  
#include "C:/prog2/data.h" // Makra, vahy a bias hodnoty
```



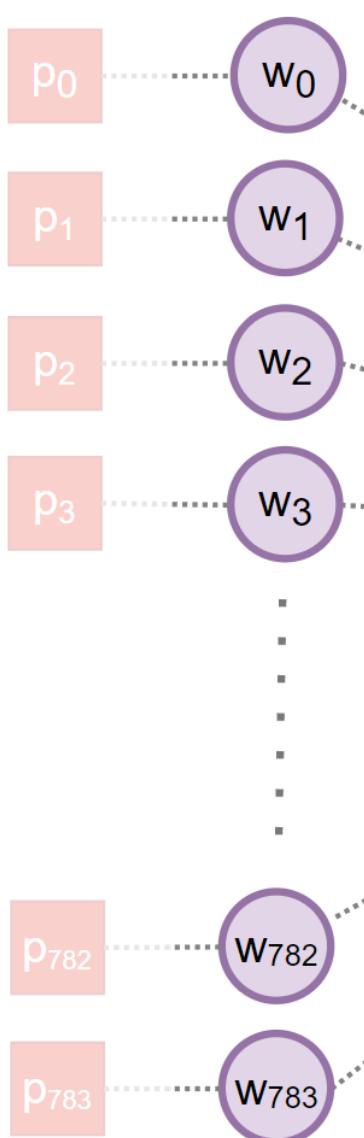
**Nesprávne** vloženie hlavičkových súborov. Tester váš kód neskompiluje.

# Režimy činnosti

- 1. Výpis váh pre zvolený neurón.**
2. Výpis váženého súčtu pre zvolený neurón a načítaný obrázok.
3. Výpis hodnoty aktivačnej funkcie ReLU pre zvolenú vstupnú hodnotu.
4. Výpis vypočítaných hodnôt funkcie Softmax.
5. Výpis indexu najväčšieho prvku v poli (Findmax).
6. Výpis načitaného obrázku číslice.
7. Výpis výsledku klasifikácie načitaného obrázku číslice.

Pixely  
obrázku

Váhy



Bias



Neurón

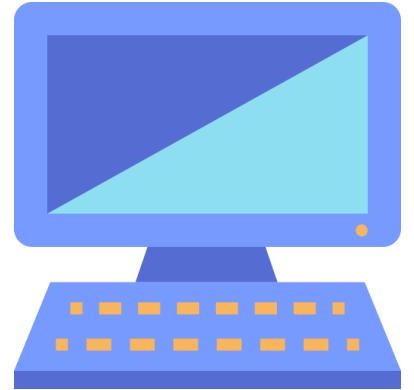


aktivačná funkcia

Aktivácia  
neurónu

# Spustenie v termináli

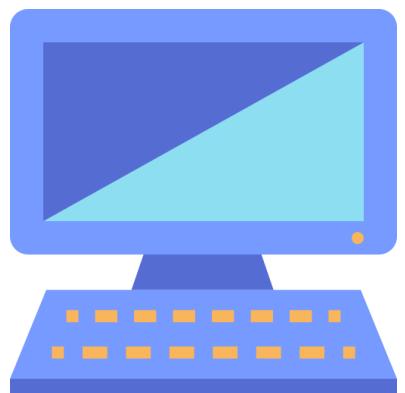
- Zadáme číslo 1.
- Následne zadáme index neurónu (0-9), ktorého váhy chceme vypísať.



C:\Users\pavol\Downloads\z2>**z2.exe**

1  
3

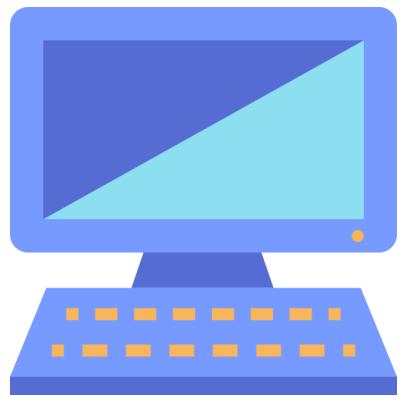
0.90	0.78	0.40	0.42	0.43	0.91	0.50	0.98	0.81	0.35
0.66	0.14	0.45	0.05	0.03	0.11	0.94	0.13	0.54	0.18
0.28	0.69	0.73	0.28	0.13	0.16	0.70	0.50	0.58	0.46
0.26	0.48	0.24	0.66	0.90	0.67	0.57	0.40	0.65	0.38
0.75	0.31	0.52	0.20	0.36	0.55	0.31	0.30	0.68	0.85
0.47	0.96	0.54	0.21	0.24	0.67	0.37	0.94	0.17	0.95
0.39	0.43	0.42	0.63	0.09	0.32	0.30	0.66	0.71	0.94
0.04	0.47	0.25	0.56	0.66	0.61	0.10	0.98	0.90	0.78
0.83	0.38	0.73	0.37	0.59	0.97	0.04	0.95	0.91	0.21
0.90	0.30	0.64	0.32	0.93	0.72	0.65	0.25	0.42	0.38
0.19	0.44	0.82	0.42	0.97	0.47	0.02	0.07	0.45	0.93
0.85	0.28	0.31	0.58	0.65	0.89	0.56	0.69	0.84	0.47
0.90	0.75	0.79	0.60	0.16	0.77	0.31	0.72	0.87	0.58
0.03	0.07	0.02	0.81	0.44	-0.00	0.34	0.51	0.10	0.80
0.44	0.95	0.09	0.75	0.54	0.75	0.68	0.17	0.48	0.53
0.65	0.47	0.38	0.45	0.92	0.28	0.94	0.02	0.87	0.77
0.63	-0.07	0.86	0.73	0.89	0.46	0.84	0.27	0.98	0.94
0.08	0.43	0.91	0.22	0.27	0.55	1.03	0.94	0.71	0.57
0.50	0.26	0.77	0.55	0.38	0.48	0.69	0.26	0.58	0.71
0.18	0.31	0.79	0.25	0.20	0.77	0.73	0.04	0.05	0.73



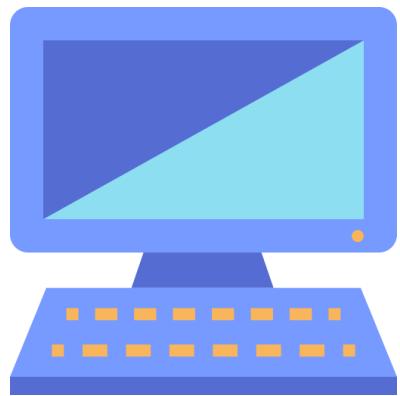
C:\Users\pavol\Downloads\  
1  
3

### Režim 1 (vstup).

0.90	0.78	0.40			0.50	0.98	0.81	0.35
0.66	0.14	0.45	0.05	0.11	0.94	0.13	0.54	0.18
0.28	0.69	0.73	0.28	0.13	0.16	0.70	0.50	0.58
0.26	0.48	0.24	0.66	0.90	0.67	0.57	0.40	0.65
0.75	0.31	0.52	0.20	0.36	0.55	0.31	0.30	0.68
0.47	0.96	0.54	0.21	0.24	0.67	0.37	0.94	0.17
0.39	0.43	0.42	0.63	0.09	0.32	0.30	0.66	0.71
0.04	0.47	0.25	0.56	0.66	0.61	0.10	0.98	0.90
0.83	0.38	0.73	0.37	0.59	0.97	0.04	0.95	0.91
0.90	0.30	0.64	0.32	0.93	0.72	0.65	0.25	0.42
0.19	0.44	0.82	0.42	0.97	0.47	0.02	0.07	0.45
0.85	0.28	0.31	0.58	0.65	0.89	0.56	0.69	0.84
0.90	0.75	0.79	0.60	0.16	0.77	0.31	0.72	0.87
0.03	0.07	0.02	0.81	0.44	-0.00	0.34	0.51	0.10
0.44	0.95	0.09	0.75	0.54	0.75	0.68	0.17	0.48
0.65	0.47	0.38	0.45	0.92	0.28	0.94	0.02	0.87
0.63	-0.07	0.86	0.73	0.89	0.46	0.84	0.27	0.98
0.08	0.43	0.91	0.22	0.27	0.55	1.03	0.94	0.71
0.50	0.26	0.77	0.55	0.38	0.48	0.69	0.26	0.58
0.18	0.31	0.79	0.25	0.20	0.77	0.73	0.04	0.05
								0.73



	Režim 1 (vstup).								
1	0.90	0.78	0.40	0.05	0.11	0.50	0.98	0.81	0.35
3	0.16	0.14	0.45	0.05	0.11	0.94	0.13	0.54	0.18
5	0.	0.69	0.73	0.28	0.13	0.16	0.70	0.50	0.58
7	0.20	0.21	0.24	0.66	0.90	0.67	0.57	0.40	0.65
9	0.7	0.20	0.36	0.55	0.31	0.30	0.68	0.85	
11	0.21	0.24	0.67	0.37	0.94	0.17	0.95		
13	0.63	0.09	0.32	0.30	0.66	0.71	0.94		
15	0.56	0.66	0.61	0.10	0.98	0.90	0.78		
17	0.37	0.59	0.97	0.04	0.95	0.91	0.21		
19	0.98	0.32	0.93	0.72	0.65	0.25	0.42	0.38	
21	0.19	0.44	0.82	0.42	0.97	0.47	0.02	0.07	0.45
23	0.85	0.28	0.31	0.58	0.65	0.89	0.56	0.69	0.47
25	0.90	0.75	0.79	0.60	0.16	0.77	0.31	0.72	0.87
27	0.03	0.07	0.02	0.81	0.44	-0.00	0.34	0.51	0.10
29	0.44	0.95	0.09	0.75	0.54	0.75	0.68	0.17	0.48
31	0.65	0.47	0.38	0.45	0.92	0.28	0.94	0.02	0.87
33	0.63	-0.07	0.86	0.73	0.89	0.46	0.84	0.27	0.98
35	0.08	0.43	0.91	0.22	0.27	0.55	1.03	0.94	0.71
37	0.50	0.26	0.77	0.55	0.38	0.48	0.69	0.26	0.58
39	0.18	0.31	0.79	0.25	0.20	0.77	0.73	0.04	0.05
41									0.73

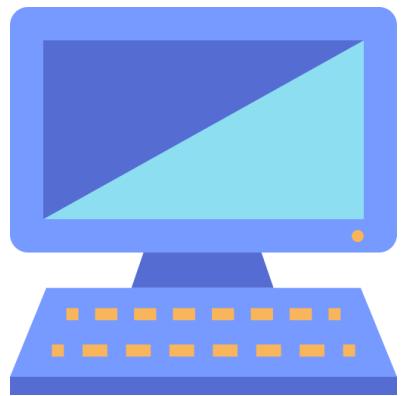


Váhy sa vypíšu  
po riadkoch.

C:\Users\pavol\Downloads\z2>z2.exe

1  
3

0.90	0.78	0.40	0.42	0.43	0.91	0.50	0.98	0.81	0.35
0.66	0.14	0.45	0.05	0.03	0.11	0.94	0.13	0.54	0.18
0.28	0.69	0.73	0.28	0.13	0.16	0.70	0.50	0.58	0.46
0.26	0.48	0.24	0.66	0.90	0.67	0.57	0.40	0.65	0.38
0.75	0.31	0.52	0.20	0.36	0.55	0.31	0.30	0.68	0.85
0.47	0.96	0.54	0.21	0.24	0.67	0.37	0.94	0.17	0.95
0.39	0.43	0.42	0.63	0.09	0.32	0.30	0.66	0.71	0.94
0.04	0.47	0.25	0.56	0.66	0.61	0.10	0.98	0.90	0.78
0.83	0.38	0.73	0.37	0.59	0.97	0.04	0.95	0.91	0.21
0.90	0.30	0.64	0.32	0.93	0.72	0.65	0.25	0.42	0.38
0.19	0.44	0.82	0.42	0.97	0.47	0.02	0.07	0.45	0.93
0.85	0.28	0.31	0.58	0.65	0.89	0.56	0.69	0.84	0.47
0.90	0.75	0.79	0.60	0.16	0.77	0.31	0.72	0.87	0.58
0.03	0.07	0.02	0.81	0.44	-0.00	0.34	0.51	0.10	0.80
0.44	0.95	0.09	0.75	0.54	0.75	0.68	0.17	0.48	0.53
0.65	0.47	0.38	0.45	0.92	0.28	0.94	0.02	0.87	0.77
0.63	-0.07	0.86	0.73	0.89	0.46	0.84	0.27	0.98	0.94
0.08	0.43	0.91	0.22	0.27	0.55	1.03	0.94	0.71	0.57
0.50	0.26	0.77	0.55	0.38	0.48	0.69	0.26	0.58	0.71
0.18	0.31	0.79	0.25	0.20	0.77	0.73	0.04	0.05	0.73



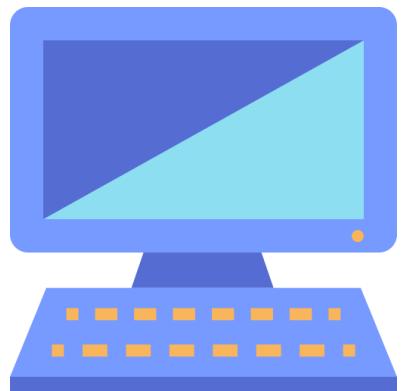
Váhy sa vypíšu  
po riadkoch.

C:\Users\pavol\Downloads\z2>z2.exe

1  
3

0.90	0.78	0.40	0.42	0.43	0.91	0.50	0.98	0.81	0.35
0.66	0.14	0.45	0.05	0.03	0.11	0.94	0.13	0.54	0.18
0.28	0.69	0.73	0.28	0.13	0.16	0.70	0.50	0.58	0.46
0.26	0.48	0.2	0.66	0.90	0.67	0.57	0.40	0.65	0.38
0.75	0.31	0.1	0.20	0.36	0.55	0.31	0.30	0.68	0.85
0.47	0.96	0	0.21	0.24	0.67	0.37	0.94	0.17	0.95
0.39	0.42	0.2	0.2	0.09	0.32	0.30	0.66	0.71	0.94
0.04	0.2	0.2	0.2	0.66	0.61	0.10	0.98	0.90	0.78
0.83	0.19	0.19	0.19	0.9	0.97	0.04	0.95	0.91	0.21
0.96	0.19	0.19	0.19	0.19	0.72	0.65	0.25	0.42	0.38
0.19	0.19	0.19	0.19	0.19	0.47	0.02	0.07	0.45	0.93
0.85	0.19	0.19	0.19	0.19	0.89	0.56	0.69	0.84	0.47
0.90	0.19	0.19	0.19	0.19	0.16	0.77	0.31	0.72	0.87
0.03	0.67	0.2	0.2	0.2	0.44	-0.00	0.34	0.51	0.10
0.44	0.95	0.09	0.75	0.54	0.75	0.68	0.17	0.48	0.53
0.65	0.47	0.38	0.45	0.92	0.28	0.94	0.02	0.87	0.77
0.63	-0.07	0.86	0.73	0.89	0.46	0.84	0.27	0.98	0.94
0.08	0.43	0.91	0.22	0.27	0.55	1.03	0.94	0.71	0.57
0.50	0.26	0.77	0.55	0.38	0.48	0.69	0.26	0.58	0.71
0.18	0.31	0.79	0.25	0.20	0.77	0.73	0.04	0.05	0.73

Každý riadok: 10  
hodnôt oddelených  
aspoň 1 medzerou.



C:\Users\pavol\Downloads\z2>z2.exe

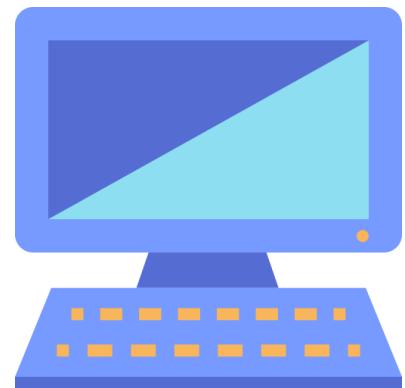
1  
3

0.90	0.78	0.40	0.42	0.43	0.91	0.50	0.98	0.81	0.35
0.66	0.14	0.45	0.05	0.03	0.11	0.94	0.13	0.54	0.18
0.28	0.69	0.73	0.28	0.13	0.16	0.70	0.50	0.58	0.46
0.26	0.48	0.2	0.66	0.90	0.67	0.57	40	0.65	0.38
0.75	0.31	0.	0.20	0.36	0.55	0.31	3	0.68	0.85
0.47	0.96	0	0.21	0.24	0.67	0.67	0.61	0.95	0.95
0.39	0.42	0.2	0.20	0.09	0.51	0.51	0.51	0.51	0.51
0.04									
0.83									
0.90									
0.19									
0.85									
0.90									
0.03	0.67	-0.07	0.86	0.73	0.44	-0.00	0.34	0.51	0.10
0.44	0.95	0.09	0.75	0.54	0.75	0.68	0.17	0.48	0.53
0.65	0.47	0.38	0.45	0.92	0.28	0.94	0.02	0.87	0.77
0.63	-0.07	0.86	0.73	0.89	0.46	0.84	0.27	0.98	0.94
0.08	0.43	0.91	0.22	0.27	0.55	1.03	0.94	0.71	0.57
0.50	0.26	0.77	0.55	0.38	0.48	0.69	0.26	0.58	0.71
0.18	0.31	0.79	0.25	0.20	0.77	0.73	0.04	0.05	0.73

Každý riadok: 10 hodnôt oddelených aspoň 1 medzerou.

Zobrazenie s presnosťou 2 desatinných miest.  
Treba použiť printf.

Váhy sa vypíšu po riadkoch.



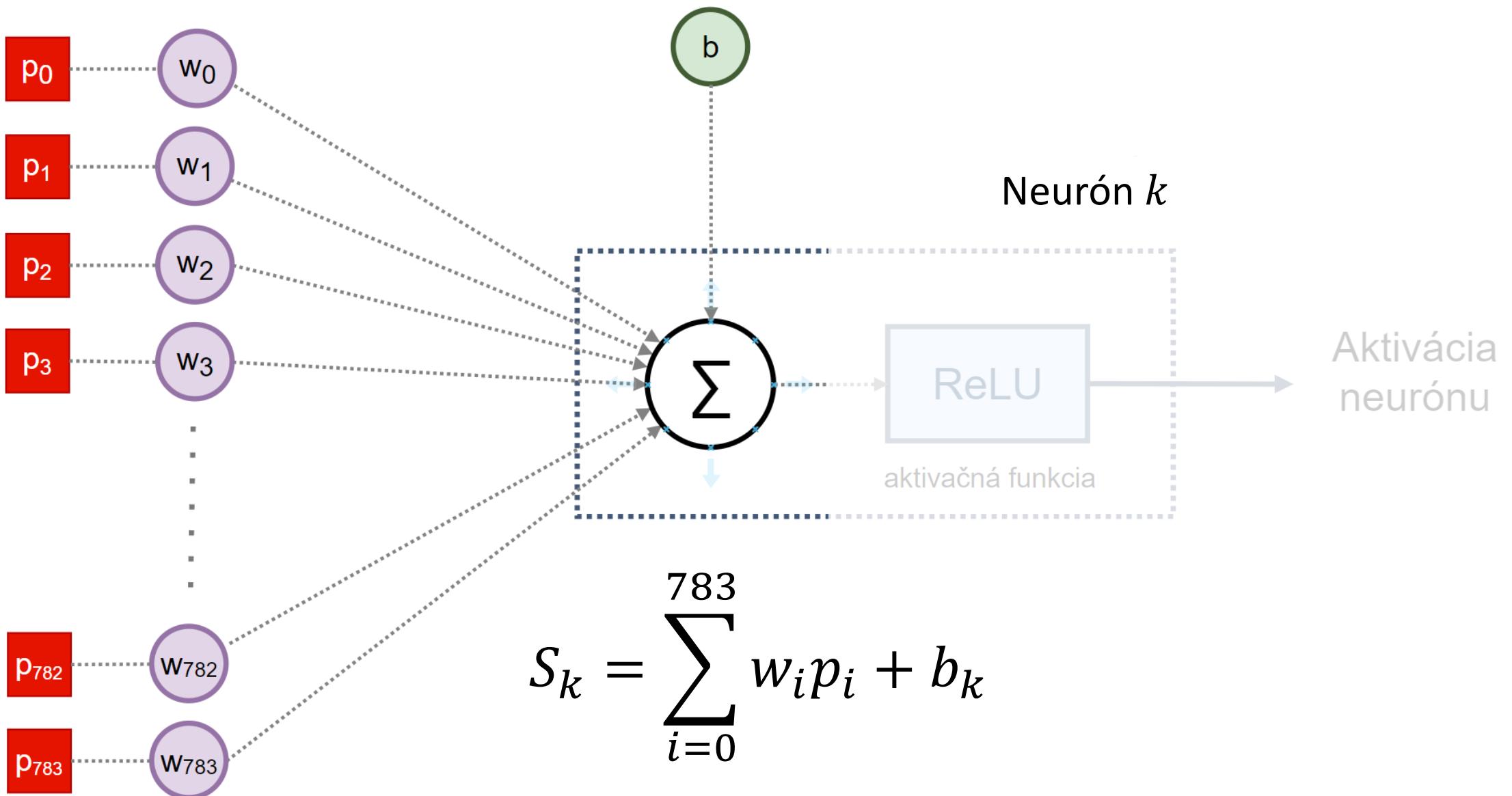
# Režimy činnosti

1. Výpis vág pre zvolený neurón.
- 2. Výpis váženého súčtu pre zvolený neurón a načítaný obrázok.**
3. Výpis hodnoty aktivačnej funkcie ReLU pre zvolenú vstupnú hodnotu.
4. Výpis vypočítaných hodnôt funkcie Softmax.
5. Výpis indexu najväčšieho prvku v poli (Findmax).
6. Výpis načitaného obrázku číslice.
7. Výpis výsledku klasifikácie načitaného obrázku číslice.

Pixely  
obrázku

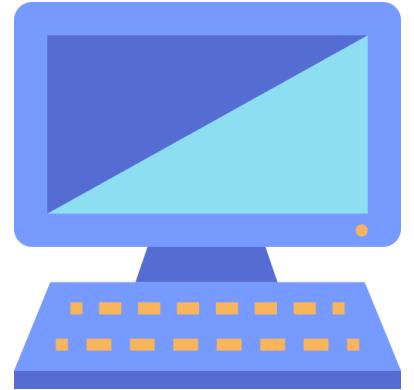
Váhy neurónu  $k$

Bias pre neurón  $k$



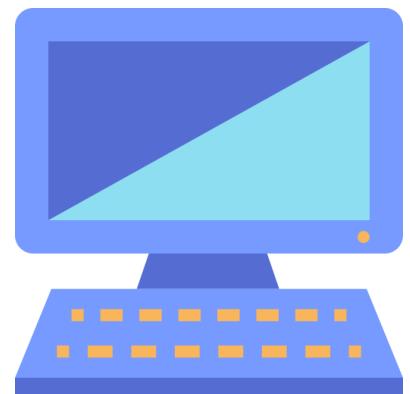
# Spustenie v termináli

- Zadáme číslo 2.
- Zadáme index neurónu (0-9).
- Zadáme pixely obrázku (784 hodnôt).

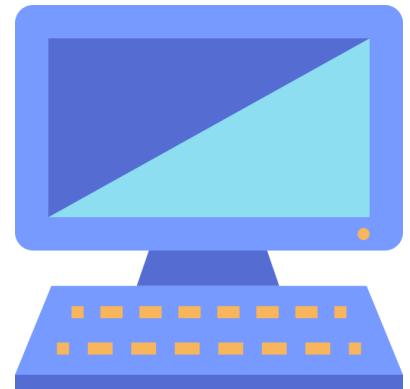


# Spustenie v termináli

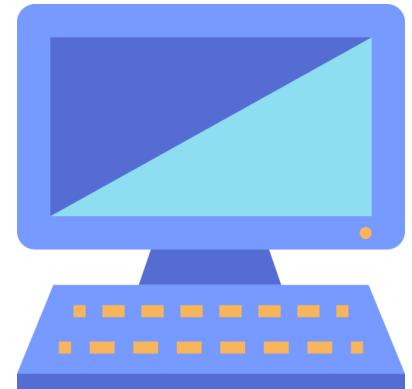
- Tento režim treba testovať pomocou presmerovania štandardného vstupu.



# Pomocný TXT súbor so všetkými vstupmi potrebnými pre režim 2



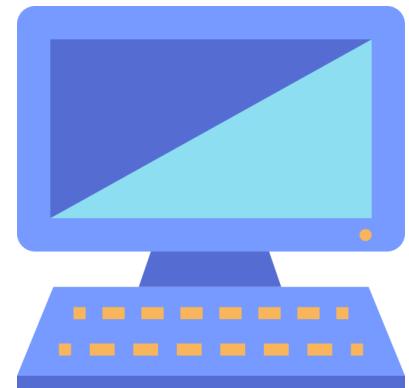
# Pomocný TXT súbor so všetkými vstupmi potrebnými pre režim 2



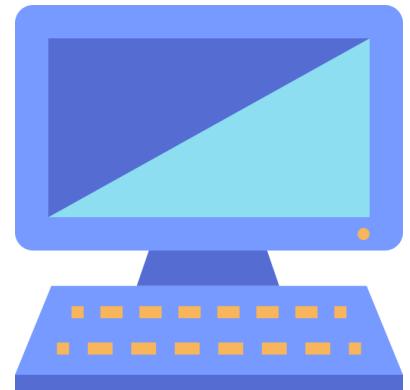
# Pomocný TXT súbor so všetkými vstupmi potrebnými pre režim 2

The screenshot shows a terminal window with the file 'mode2.txt' open. The window title is 'mode2.txt'. The menu bar includes 'File', 'Edit', 'View', and a settings gear icon. The main content displays two large blue callout bubbles. The top bubble contains the text 'Režim 2 (vstup)' and points to the first few lines of numerical data representing the input layer. The bottom bubble contains the text 'Neurón indexom 7 (vstup)' and points to the data for the 7th neuron. The data consists of two columns of numbers, representing the input values for each of the 10 neurons.

Neuron	Value
1	0.00 0.00 0.00
2	0.00 0.00 0.00
3	0.00 0.00 0.00
4	0.00 0.00 0.00
5	0.00 0.00 0.00
6	0.00 0.00 0.00
7	0.00 0.00 0.00
8	0.00 0.00 0.00
9	0.00 0.00 0.00
10	0.00 0.00 0.00

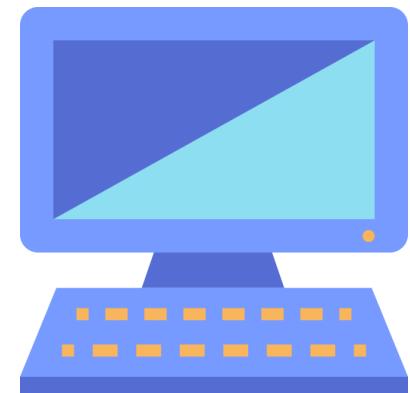


# Pomocný TXT súbor so všetkými vstupmi potrebnými pre režim 2



Program spustíme s presmerovaním  
štandardného vstupu zo súboru.

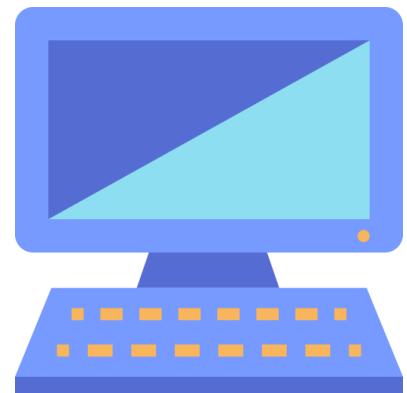
```
C:\Users\pavol\Downloads\z2>z2.exe < mode2.txt
56.38
C:\Users\pavol\Downloads\z2>[]
```



Program spustíme s presmerovaním  
štandardného vstupu zo súboru.

```
C:\Users\pavol\Downloads\z2>z2.exe < mode2.txt
56.38
C:\Users\pavol\Downloads\z2>[]
```

Vážený súčet sa vypíše s  
presnosťou 2 desatinných  
miest. Treba použiť printf.



# Režimy činnosti

1. Výpis váh pre zvolený neurón.
2. Výpis váženého súčtu pre zvolený neurón a načítaný obrázok.
- 3. Výpis hodnoty aktivačnej funkcie ReLU pre zvolenú vstupnú hodnotu.**
4. Výpis vypočítaných hodnôt funkcie Softmax.
5. Výpis indexu najväčšieho prvku v poli (Findmax).
6. Výpis načitaného obrázku číslice.
7. Výpis výsledku klasifikácie načitaného obrázku číslice.

Pixely  
obrázku

Váhy

Bias

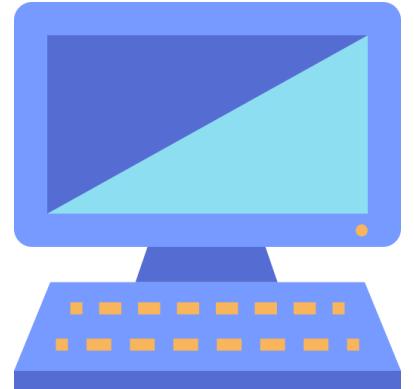
Neurón

Aktivácia  
neurónu

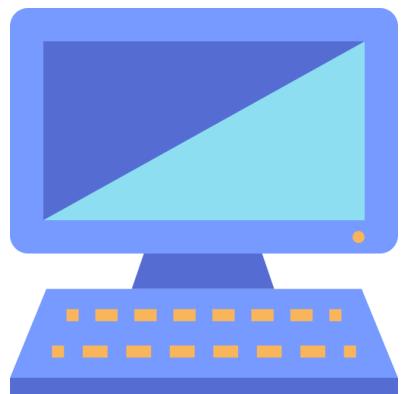


# Spustenie v termináli

- Zadáme číslo 3.
- Zadáme hodnotu vstupu do ReLU funkcie.



```
C:\Users\pavol\Downloads\z2>z2.exe  
3  
0.24  
0.24  
C:\Users\pavol\Downloads\z2>
```



Režim 3 (vstup).

```
C:\Users\pavol\Downloads\z2>z2.exe
3
0.24
0.24
C:\Users\pavol\Downloads\z2>
```



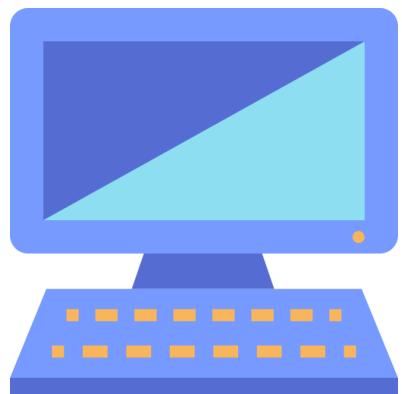
Režim 3 (vstup).

```
C:\Users\pavol>3
0.24
0.24
C:\Users\pavol\Downloads\z2>
```

Hodnota pre  
ReLU funkciu  
(vstup).



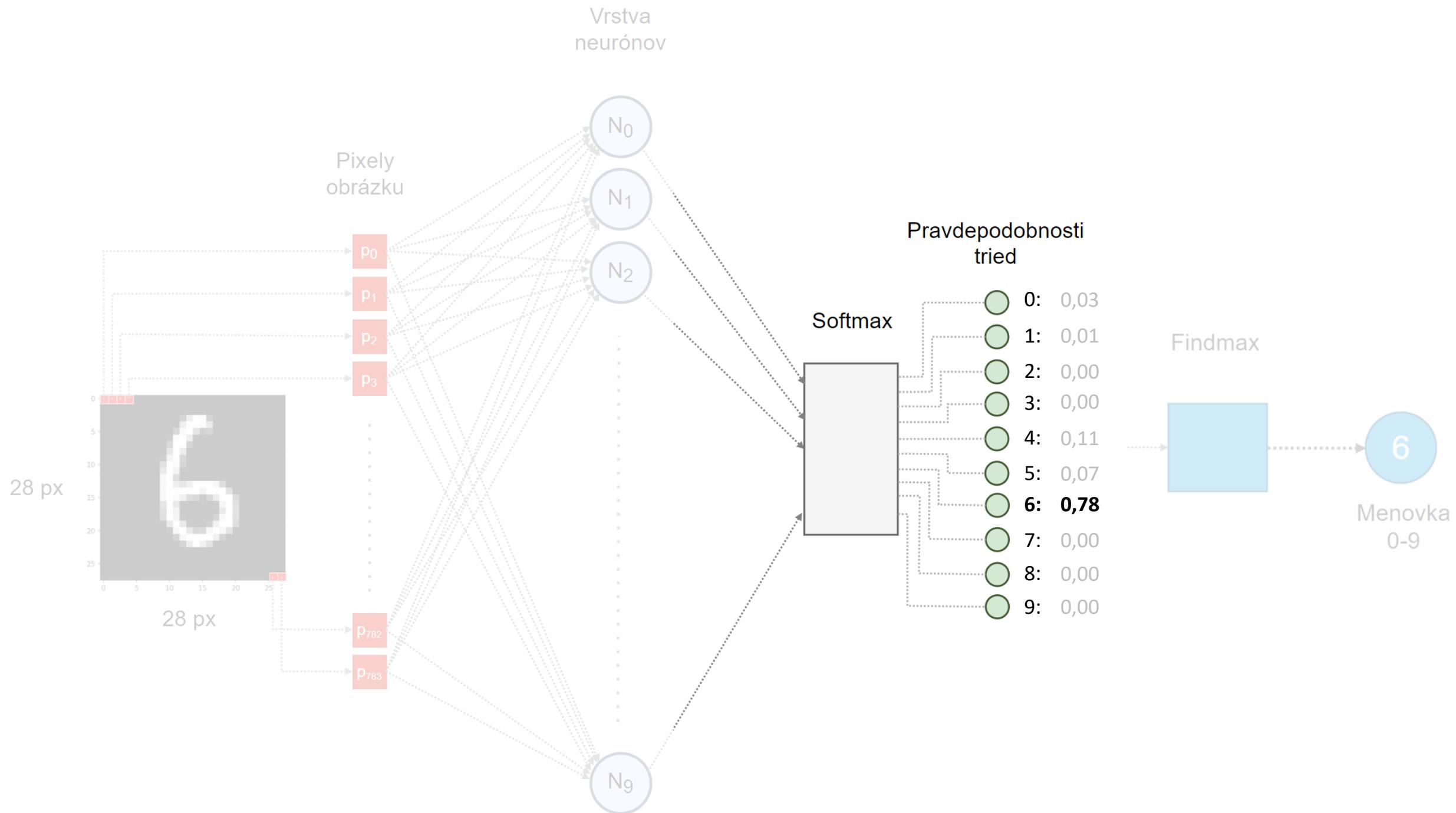
```
C:\Users\pavol\Downloads\z2>z2.exe
3
0.24
0.24
C:\Users\pavol\Downloads\z2>
```



Vypočítaná hodnota ReLU funkcie sa zobrazí s presnosťou 2 desatinných miest. Treba použiť printf.

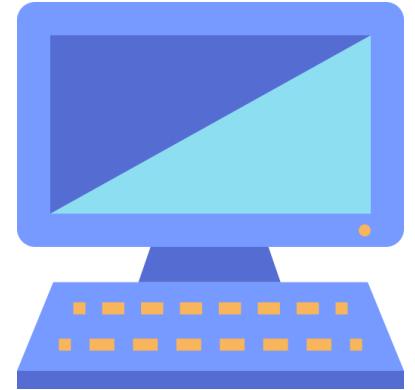
# Režimy činnosti

1. Výpis váh pre zvolený neurón.
2. Výpis váženého súčtu pre zvolený neurón a načítaný obrázok.
3. Výpis hodnoty aktivačnej funkcie ReLU pre zvolenú vstupnú hodnotu.
- 4. Výpis vypočítaných hodnôt funkcie Softmax.**
5. Výpis indexu najväčšieho prvku v poli (Findmax).
6. Výpis načitaného obrázku číslice.
7. Výpis výsledku klasifikácie načitaného obrázku číslice.

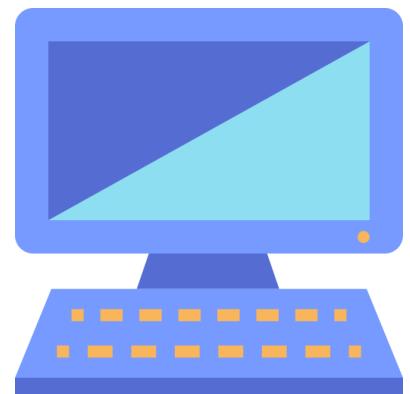


# Spustenie v termináli

- Zadáme číslo 4.
- Zadáme 10 čísiel reprezentujúcich pole, ktoré odovzdáme do funkcie Softmax.

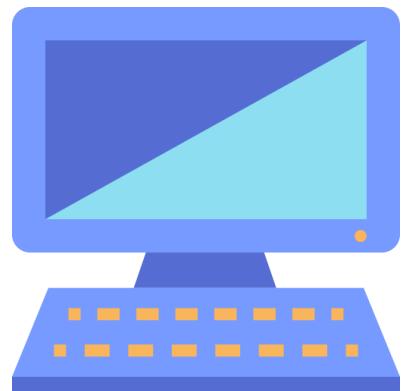


```
C:\Users\pavol\Downloads\z2>z2.exe  
4  
2.03 0.03 0.45 1.14 -0.36 5.13 4.89 -20.15 3.08 4.12  
0.02 0.00 0.00 0.01 0.00 0.42 0.33 0.00 0.05 0.15  
C:\Users\pavol\Downloads\z2>[]
```



Režim 4 (vstup).

```
C:\Users\pavol\Downloads\z2>z2.exe
4
2.03 0.03 0.45 1.14 -0.36 5.13 4.89 -20.15 3.08 4.12
0.02 0.00 0.00 0.01 0.00 0.42 0.33 0.00 0.05 0.15
C:\Users\pavol\Downloads\z2>[]
```



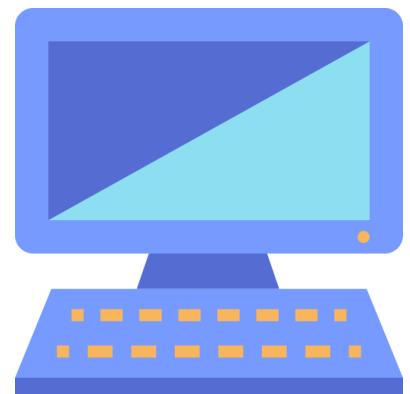
Režim 4 (vstup).

10 hodnôt  
vstupného poľa pre  
funkciu Softmax  
(vstup).

```
C:\Users\pavol\Downloads\z2>z2.exe
4
2.03 0.03 0.45 1.14 -0.36 5.13 4.89 -20.15 3.08 4.12
0.02 0.00 0.00 0.01 0.00 0.42 0.33 0.00 0.05 0.15
C:\Users\pavol\Downloads\z2>[]
```



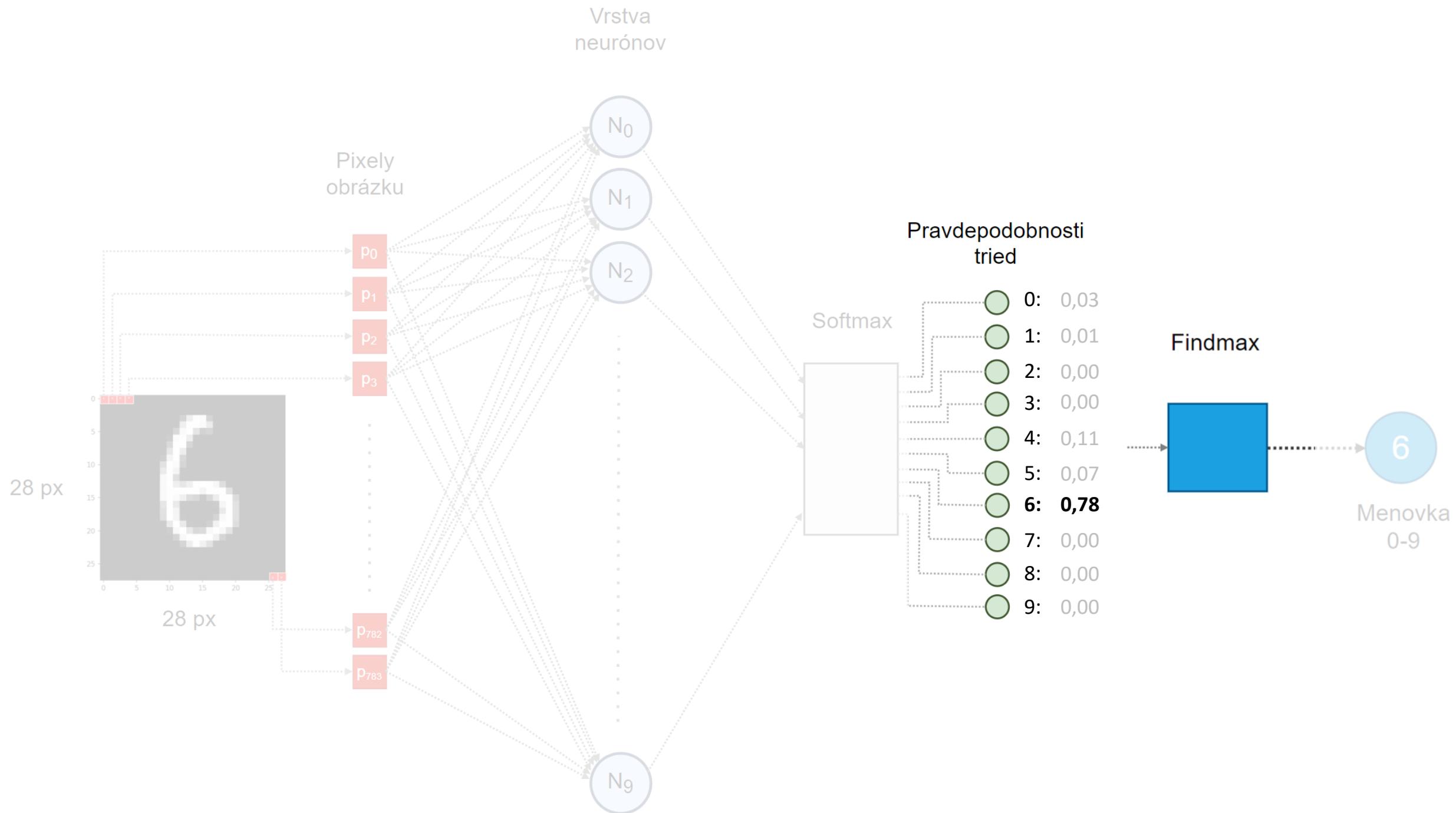
```
C:\Users\pavol\Downloads\z2>z2.exe
4
2.03 0.03 0.45 1.14 -0.36 5.13 4.89 -20.15 3.08 4.12
0.02 0.00 0.00 0.01 0.00 0.42 0.33 0.00 0.05 0.15
C:\Users\pavol\Downloads\z2>[
```



Výstup funkcie Softmax (10 hodnôt pravdepodobnosti).  
Presnosť 2 desatinné miesta.  
Treba použiť printf.

# Režimy činnosti

1. Výpis vág pre zvolený neurón.
2. Výpis váženého súčtu pre zvolený neurón a načítaný obrázok.
3. Výpis hodnoty aktivačnej funkcie ReLU pre zvolenú vstupnú hodnotu.
4. Výpis vypočítaných hodnôt funkcie Softmax.
- 5. Výpis indexu najväčšieho prvku v poli (Findmax).**
6. Výpis načitaného obrázku číslice.
7. Výpis výsledku klasifikácie načitaného obrázku číslice.

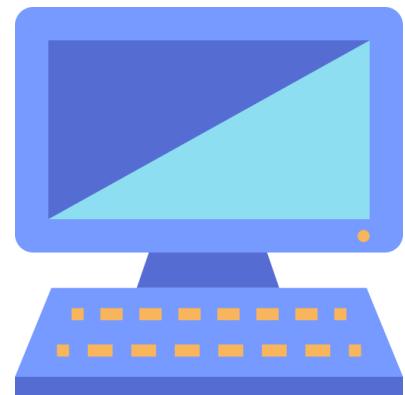


# Spustenie v termináli

- Zadáme číslo 5.
- Zadáme 10 čísiel reprezentujúcich pole, ktoré odovzdáme do funkcie Findmax.

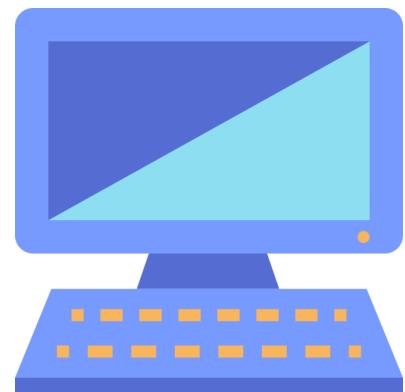


```
C:\Users\pavol\Downloads\z2>z2.exe
5
0.02 0.00 0.66 0.00 0.00 0.04 0.00 0.13 0.00 0.15
2
C:\Users\pavol\Downloads\z2>[]
```



Režim 5 (vstup).

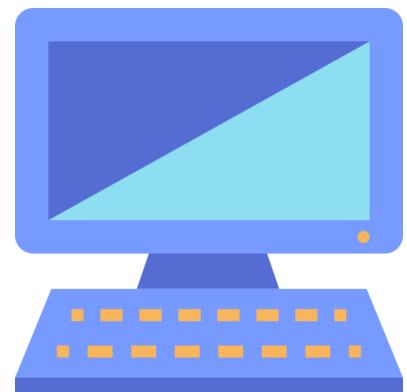
```
C:\Users\pavol\Downloads\z2>z2.exe
5
0.02 0.00 0.66 0.00 0.00 0.04 0.00 0.13 0.00 0.15
2
C:\Users\pavol\Downloads\z2>[]
```



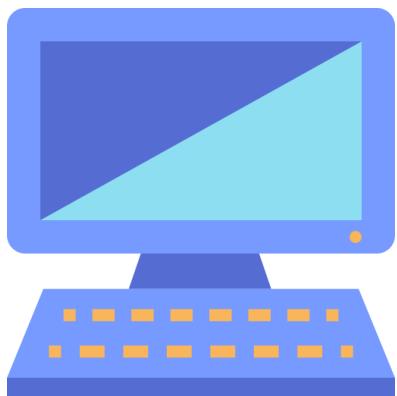
Režim 5 (vstup).

10 hodnôt  
vstupného poľa pre  
funkciu Findmax  
(vstup).

```
C:\Users\pavol\Downloads\z2>z2.exe
5
0.02 0.00 0.66 0.00 0.00 0.04 0.00 0.13 0.00 0.15
2
C:\Users\pavol\Downloads\z2>[]
```



```
C:\Users\pavol\Downloads\z2>z2.exe
5
0.02 0.00 0.66 0.00 0.00 0.04 0.00 0.13 0.00 0.15
2
C:\Users\pavol\Downloads\z2>[]
```



Vypíše sa index  
najväčšieho prvku v  
poli.

# Režimy činnosti

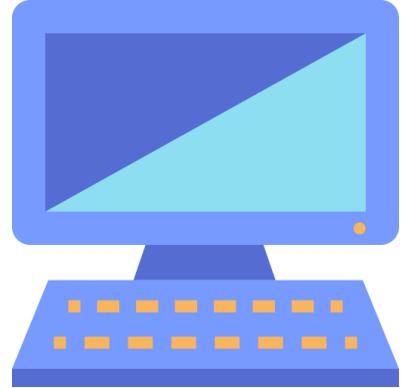
1. Výpis váh pre zvolený neurón.
2. Výpis váženého súčtu pre zvolený neurón a načítaný obrázok.
3. Výpis hodnoty aktivačnej funkcie ReLU pre zvolenú vstupnú hodnotu.
4. Výpis vypočítaných hodnôt funkcie Softmax.
5. Výpis indexu najväčšieho prvku v poli (Findmax).
- 6. Výpis načitaného obrázku číslice.**
7. Výpis výsledku klasifikácie načitaného obrázku číslice.

Výpis načítaného obrázku do konzoly pomocou dodanej funkcie *print\_image()*.

```
.....  
.....  
.....  
.....WWWWW.....  
.....WWW..WWWWWWWW.....  
.....WW.....WWW.....  
.....WW.....WWW.....  
.....WW.....WW.....  
.....WW.....WW.....  
.....WWW.....WWW.....  
.....WW.....WWW.....  
.....WWW.....WW.....  
.....WWWW.....WW.....  
.....WWW.....WW.....  
.....WWWW.....WW.....  
.....WW.....WW.....  
.....WW.....WW.....  
.....WWWW.....WWWW.....  
.....WWWWW.....
```

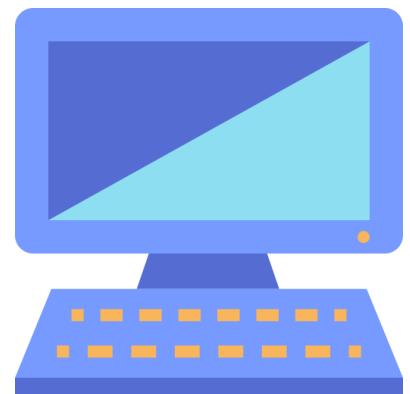
# Spustenie v termináli

- Zadáme číslo 6.
- Zadáme pixely obrázku (784 hodnôt).



# Spustenie v termináli

- Tento režim treba testovať pomocou presmerovania štandardného vstupu.







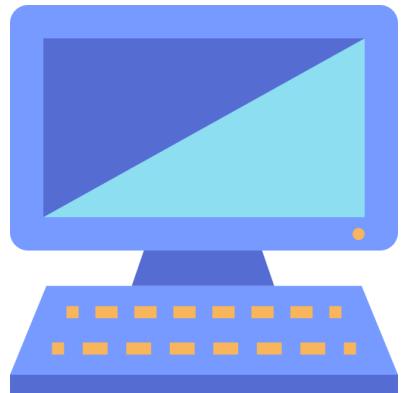


Program  
spustíme s  
presmerovaním  
štandardného  
vstupu zo súboru.

```
C:\Users\pavol\Downloads\z2>z2.exe < mode6.txt
```

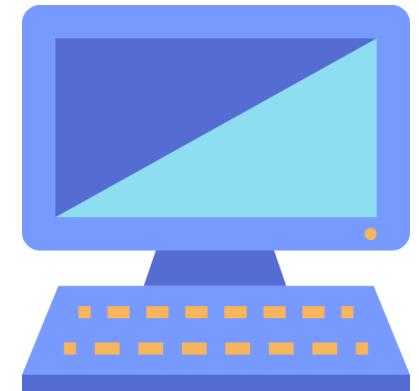
```
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....
```



Program  
spustíme s  
presmerovaním  
štandardného  
vstupu zo súboru.

```
C:\Users\pavol\Downloads\z2>z2.exe < mode6.txt
```

Výpis obrázku  
pomocou dodanej  
funkcie print\_image()

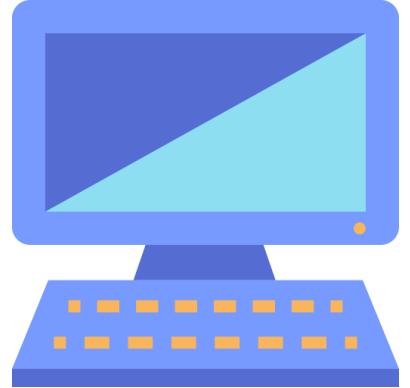


# Režimy činnosti

1. Výpis vág pre zvolený neurón.
2. Výpis váženého súčtu pre zvolený neurón a načítaný obrázok.
3. Výpis hodnoty aktivačnej funkcie ReLU pre zvolenú vstupnú hodnotu.
4. Výpis vypočítaných hodnôt funkcie Softmax.
5. Výpis indexu najväčšieho prvku v poli (Findmax).
6. Výpis načitaného obrázku číslice.
7. **Výpis výsledku klasifikácie načitaného obrázku číslice.**

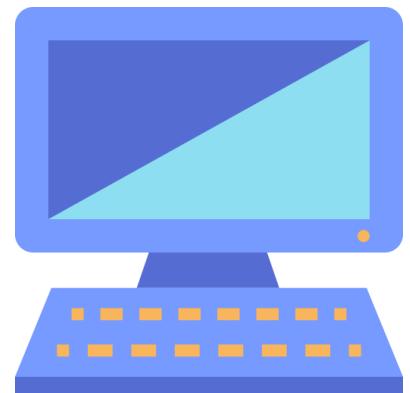
# Spustenie v termináli

- Zadáme číslo 7.
- Zadáme pixely obrázku (784 hodnôt).



# Spustenie v termináli

- Tento režim treba testovať pomocou presmerovania štandardného vstupu.

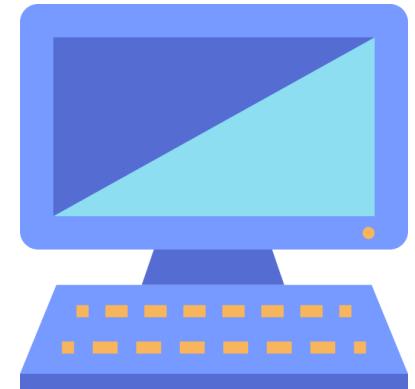




# Pomocný TXT súbor so všetkými vstupmi potrebnými pre režim 7

```
mode7.txt
File Edit View
7
0.00 b
0.00 0.
0.00 0.0.
0.00 0.00 b.
0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 b.
```

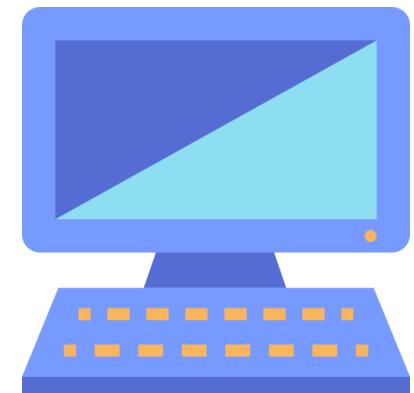
Režim 7 (vstup).





Program spustíme s presmerovaním  
štandardného vstupu zo súboru.

```
C:\Users\pavol\Downloads\z2>z2.exe < mode7.txt
3
C:\Users\pavol\Downloads\z2>
```



Program spustíme s presmerovaním  
štandardného vstupu zo súboru.

```
C:\Users\pavol\Downloads\z2>z2.exe < mode7.txt
3
C:\Users\pavol\Downloads\z2>
```

Výsledok klasifikácie  
obrázku (číslo 0-9).



# Ako načítať obrázok?



- V tomto zadaní je potrebné načítavať obrázky zo štandardného vstupu (stdin).
- Každý obrázok obsahuje 784 pixelov (28x28).
- Každý pixel má hodnotu v rozsahu  $\langle 0,1 \rangle$ .
- Na načítanie obrázku (všetkých 784 pixelov) stačí použiť dodanú funkciu `load_data()`.
- Táto funkcia je deklarovaná v súbore `functions.h`.

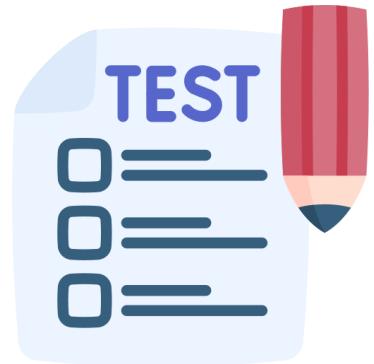
# Ako načítať obrázok?



```
// funkcia na nacitanie hodnot pola 'data' s dlzkou 'len'  
// zo standardneho vstupu (stdin)  
void load_data(double *data, const unsigned int len){  
    for(unsigned int i = 0; i< len;i++){  
        scanf( format: "%lf", &data[i]);  
    }  
}
```

# Ako testovať váš program?

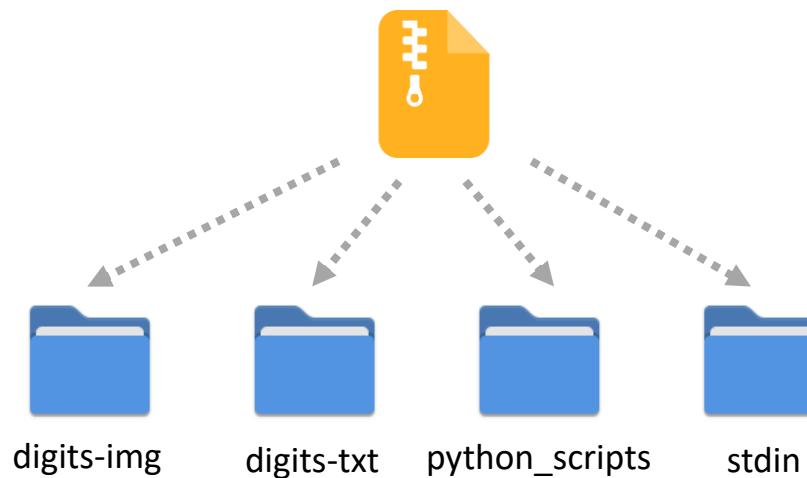
- Spolu s projektom máte k dispozícii priečinok s testovacími súbormi.



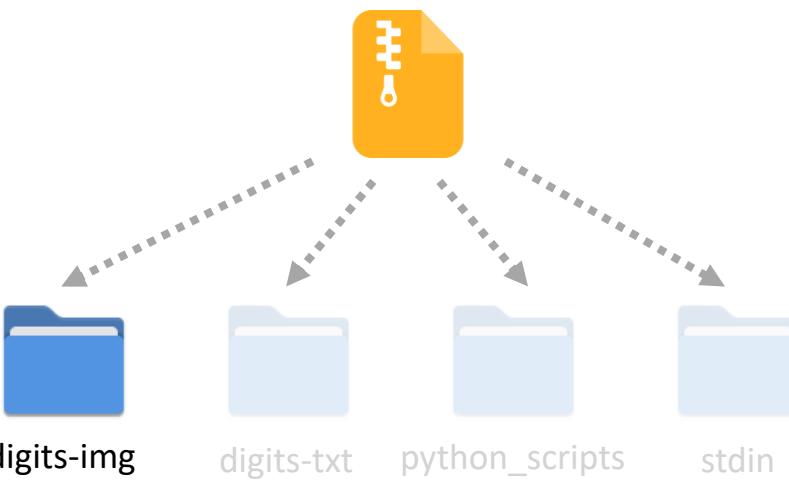
`z2_testing.zip`



`z2_testing.zip`



## `z2_testing.zip`



0_0.png	0_1.png	0_2.png	1_0.png	1_1.png	1_2.png

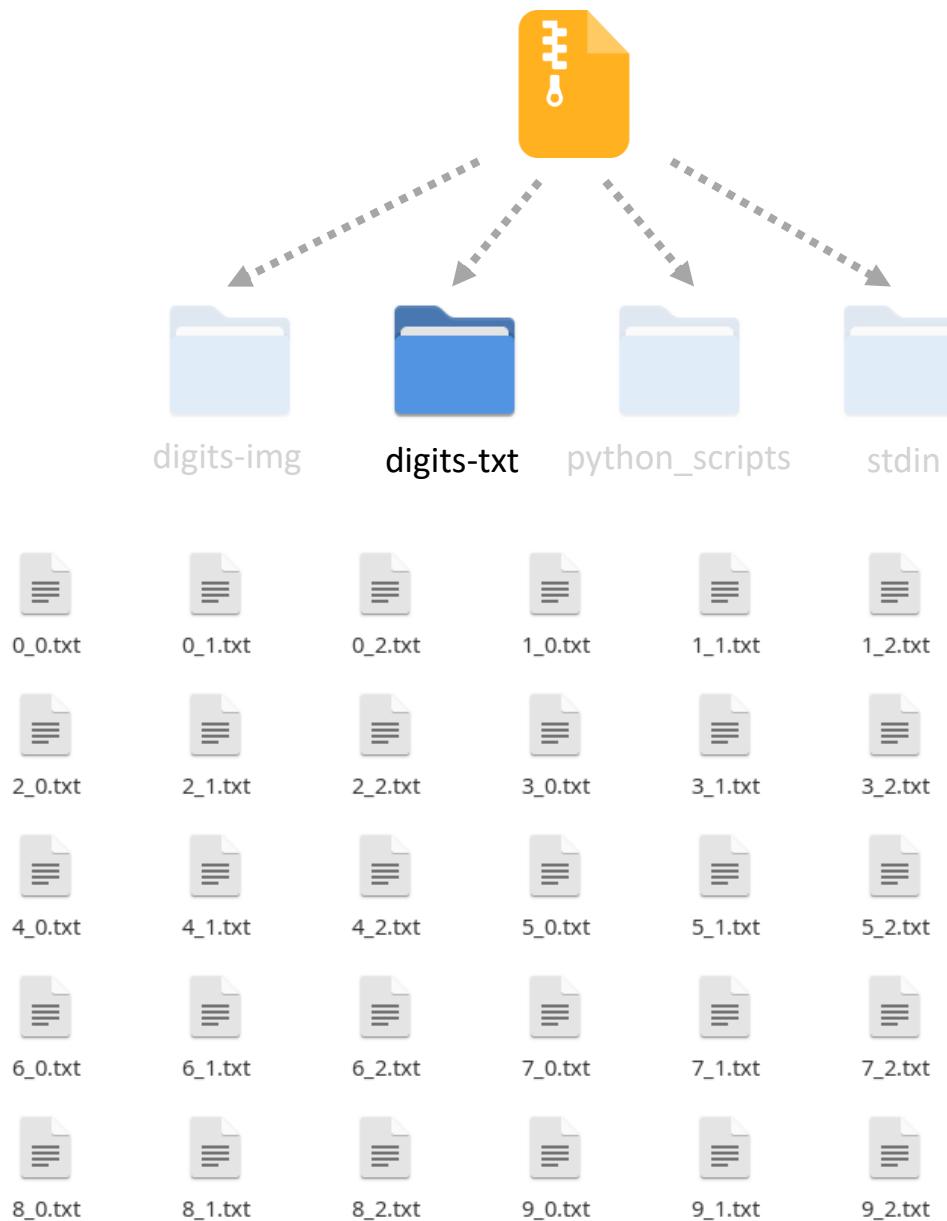
2_0.png	2_1.png	2_2.png	3_0.png	3_1.png	3_2.png

4_0.png	4_1.png	4_2.png	5_0.png	5_1.png	5_2.png

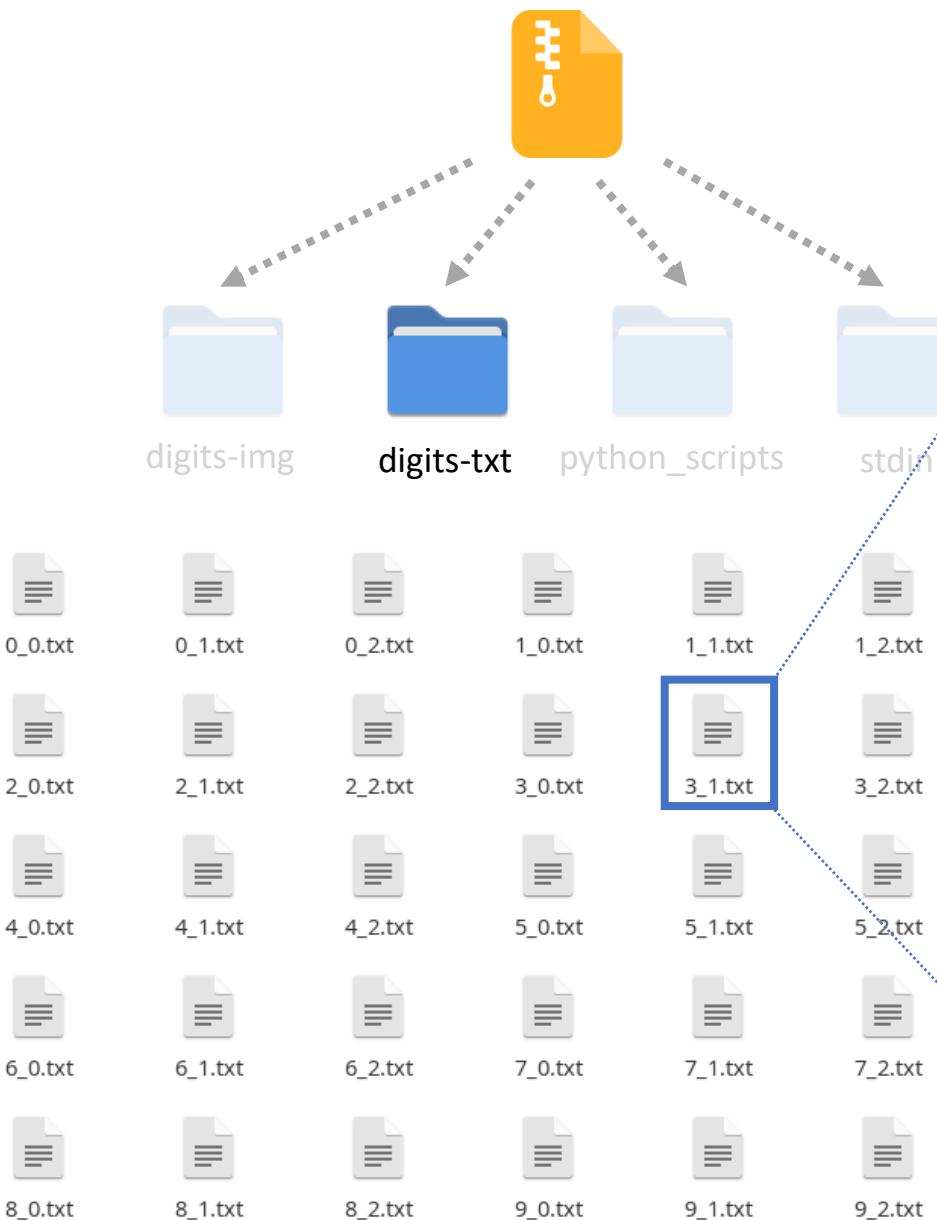
6_0.png	6_1.png	6_2.png	7_0.png	7_1.png	7_2.png

8_0.png	8_1.png	8_2.png	9_0.png	9_1.png	9_2.png

## `z2_testing.zip`



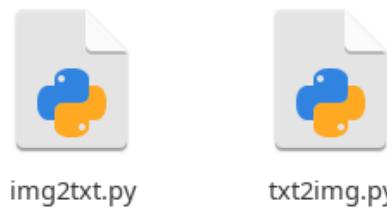
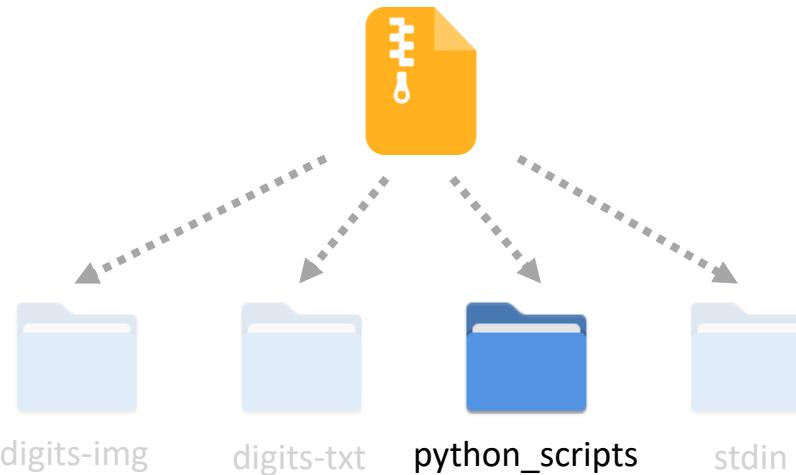
## **z2\_testing.zip**



## **TXT reprezentácia obrázku**

- 784 hodnôt pixelov
  - oddelené medzerou
  - pixely sú v rozsahu  $\langle 0,1 \rangle$

`z2_testing.zip`



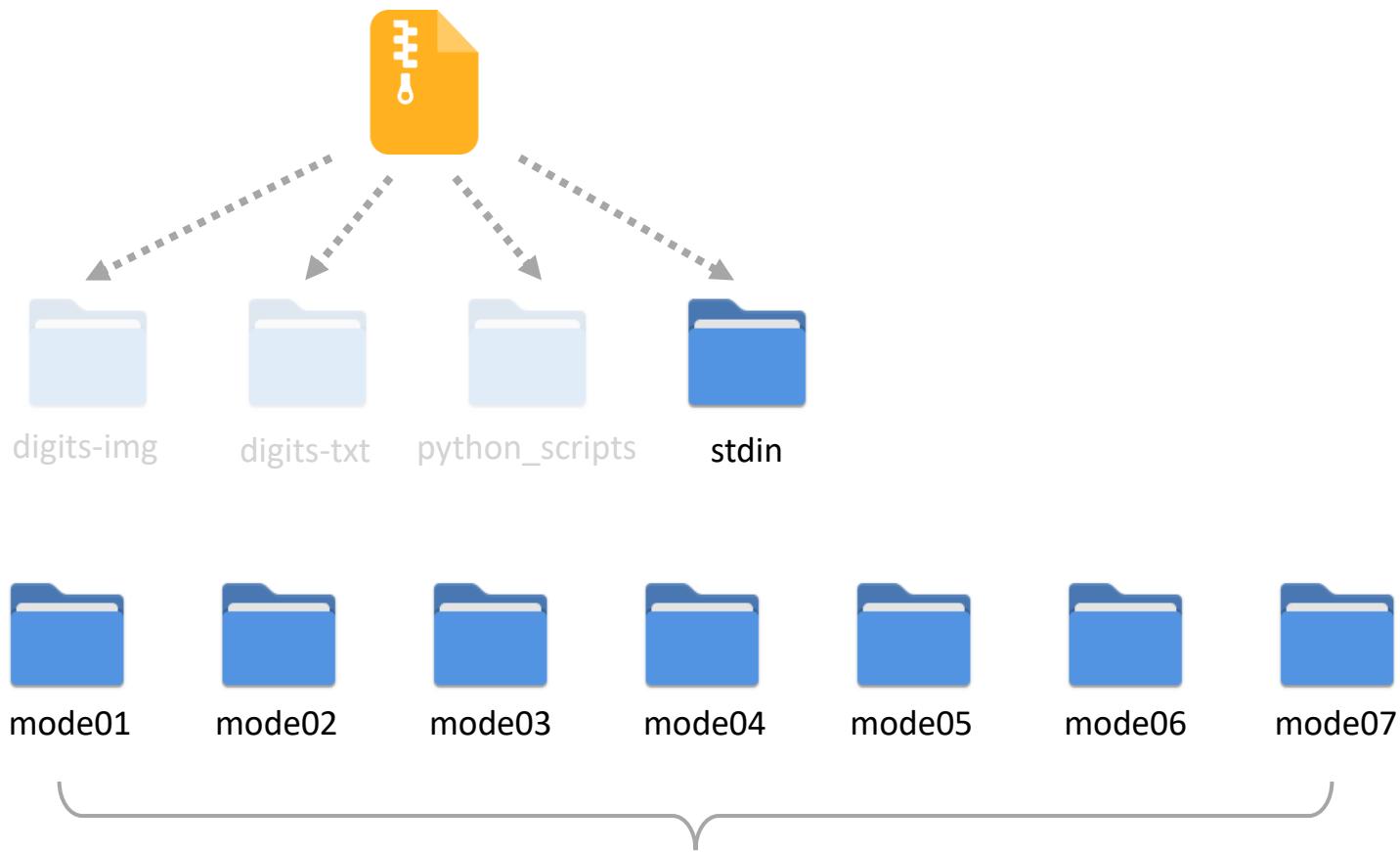
Python skripty na dobrovoľné  
experimentovanie.

# Pomocné Python skripty – na experimenty

S projektom sú dodané 2 Python skripty, ktoré slúžia na vlastné dobrovoľné experimenty:

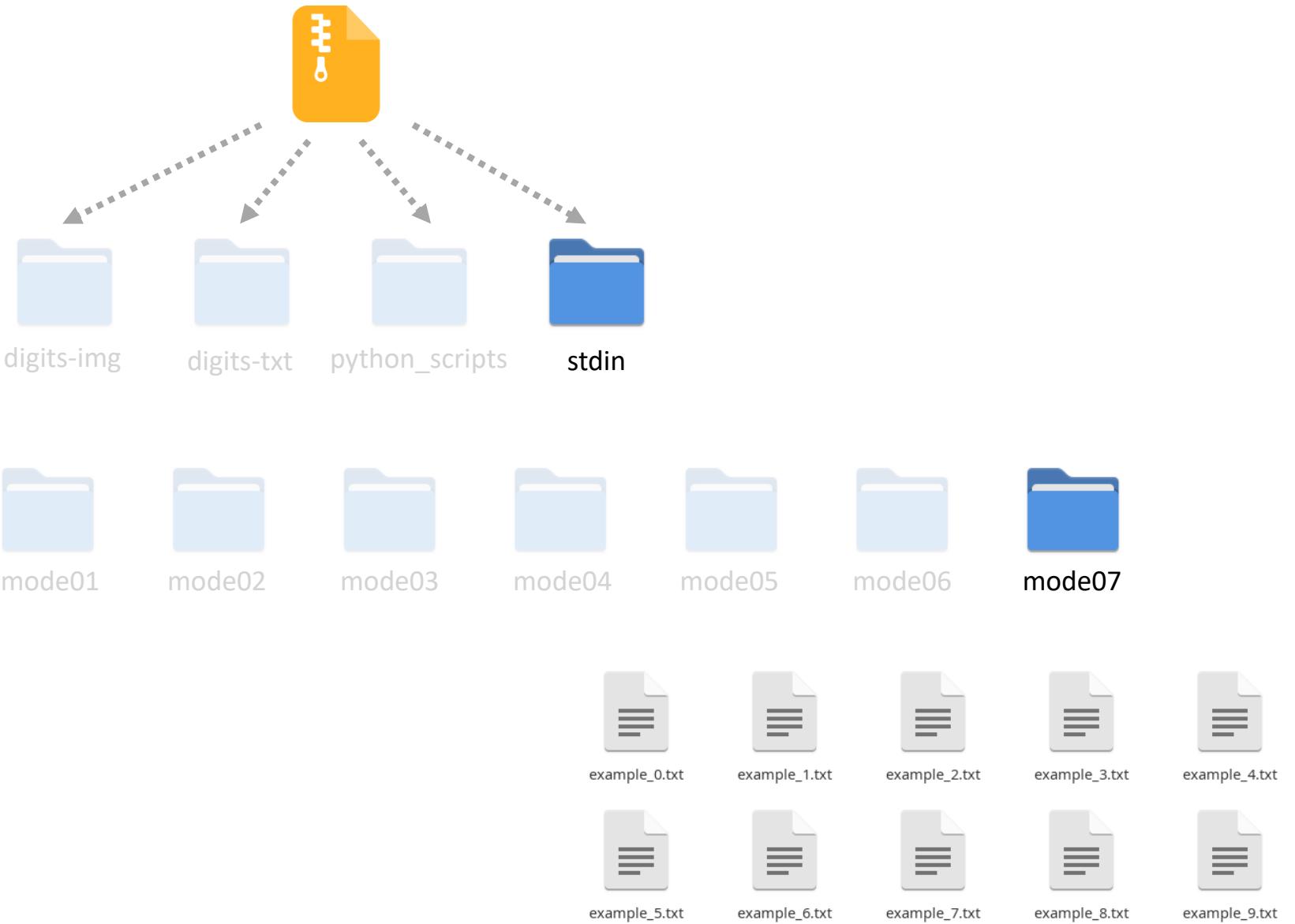
- **img2txt.py** – prevod grayscale obrázku do TXT podoby (uložia sa hodnoty pixelov v rozsahu  $\langle 0,1 \rangle$ )
- **txt2img.py** – prevod obrázku v TXT súbore do vizuálnej podoby (očakáva sa, že pixely sú v TXT súbore v rozsahu  $\langle 0,1 \rangle$ )

## `z2_testing.zip`



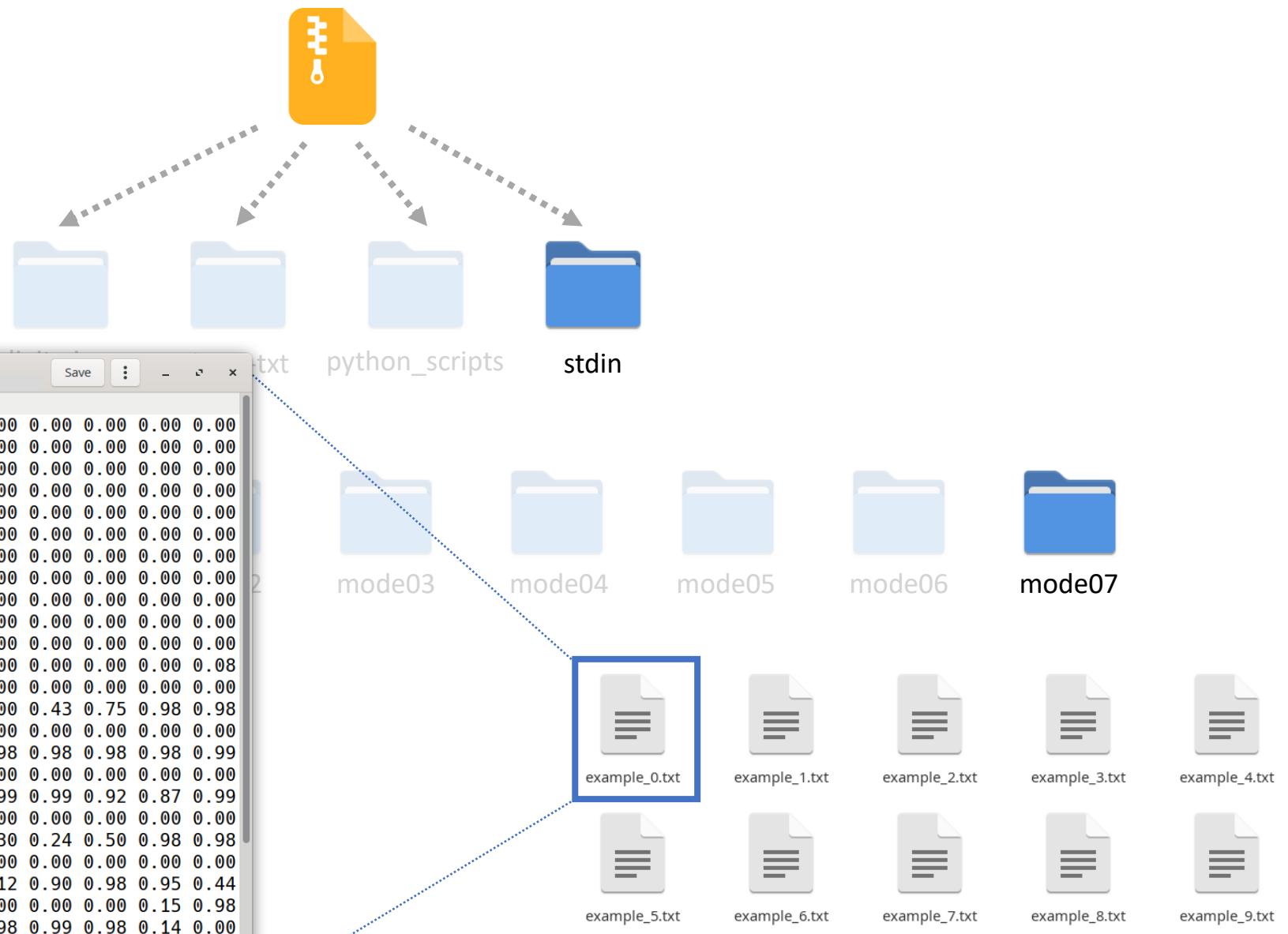
Testovacie vstupy pre  
jednotlivé režimy činnosti

## `z2_testing.zip`



## z2\_testing.zip

TXT súbory obsahujú vstupy do programu (stdin). Pri spustení programu v termináli je potrebné presmerovať stdin z daného súboru.



# Chybové situácie

- V tomto zadaní netreba ošetrovať vstupné hodnoty.
- Program bude automatizované testovaný len s platnými a zmysluplnými hodnotami.



# Bodovanie

Testovacie scenáre		
Scenár 1	Režim 1 - výpis váh	1,0 b
Scenár 2	Režim 2 - výpis váženého súčtu	2,0 b
Scenár 3	Režim 3 – výpis hodnoty ReLU funkcie	0,5 b
Scenár 4	Režim 4 – výpis hodnôt funkcie Softmax	1,5 b
Scenár 5	Režim 5 – výpis funkcie Findmax	0,5 b
Scenár 6	Režim 6 – výpis načítaného obrázku	0,5 b
Scenár 7	Režim 7 – výpis výsledku klasifikácie načítaného obrázku	4,0 b
Súčet		10 b



Živá ukážka

# Literatúra

- [https://en.wikipedia.org/wiki/MNIST\\_database](https://en.wikipedia.org/wiki/MNIST_database)
- <https://slovnik.juls.savba.sk/>
- <https://korektor.sk/editor/info>
- <https://app.diagrams.net/>
- <https://me-qr.com/qr-code-generator/qr>

# Zdroje obrázkov

- <https://www.flaticon.com>
- <https://www.freepik.com>