

O projektu

Neuronová síť

Algoritmus

Přizpůsobení

Testování

Flappy Bird

Se strojovým učením

Roční projekt NPRX035 / NPROG035

Autor - Saydametov Nikita

Začít poznávat



O projektu

Podstata projektu



Flappy Bird

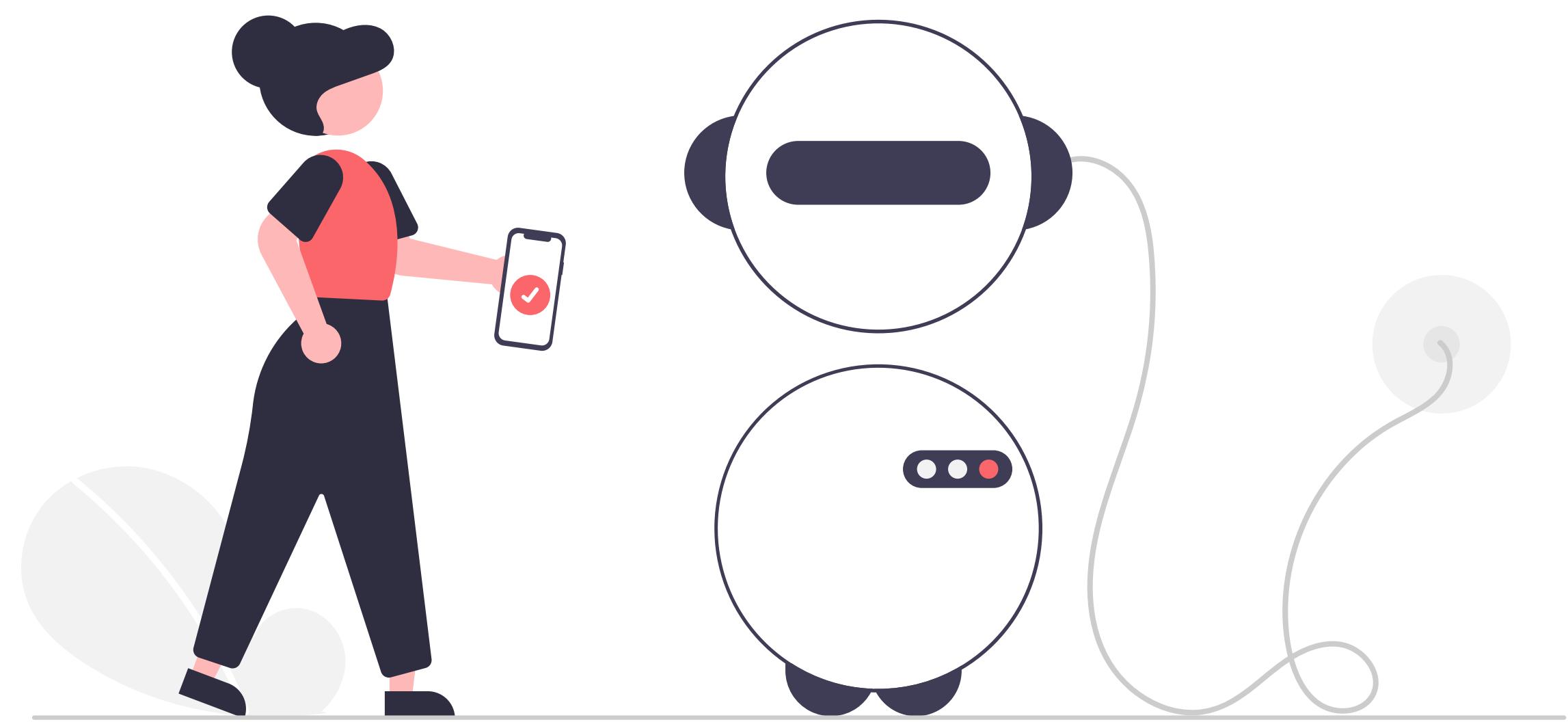
Tento projekt je herní ovladač umělé inteligence založený na neurosetích a genetickém algoritmu pro hru **Flappy Bird**

O projektu

Algoritmus strojového učení

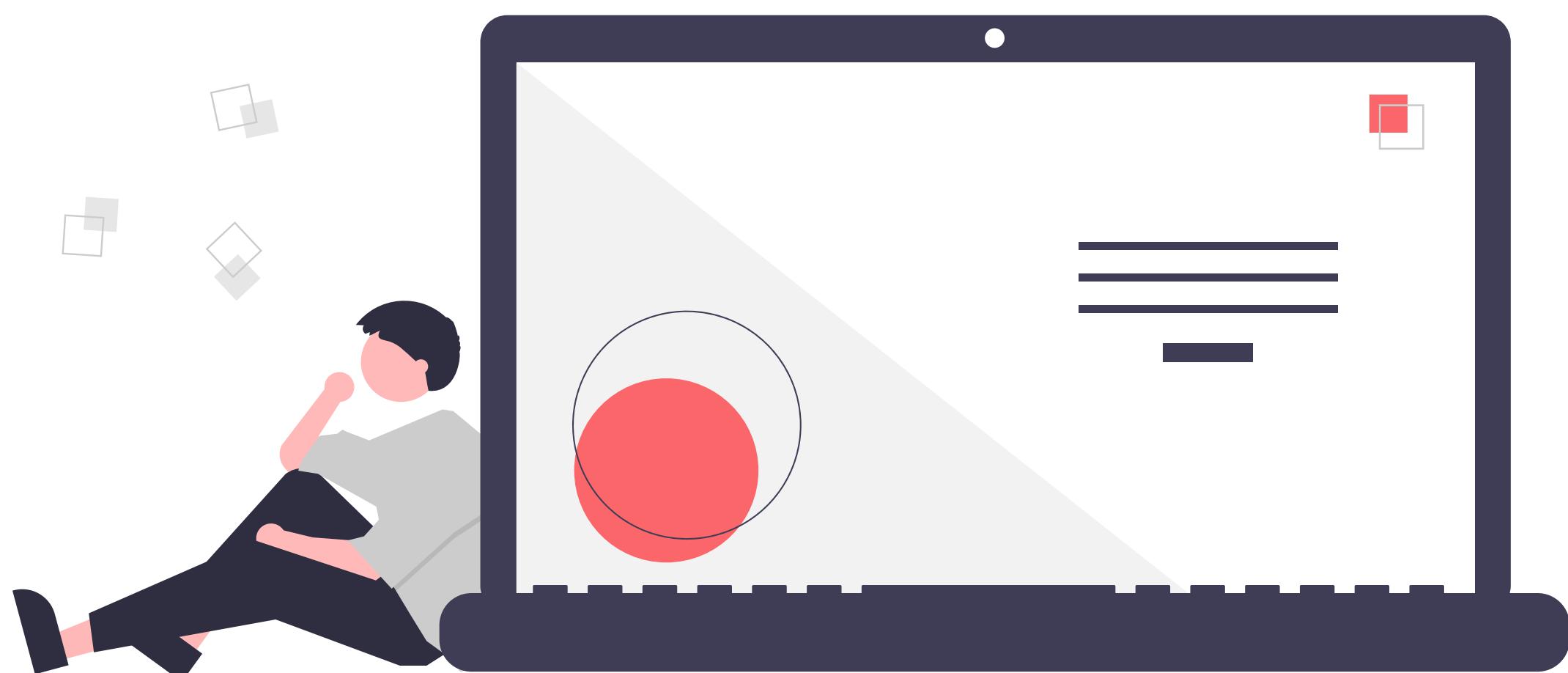
Podle formulace **Arthuria Samuela** z roku 1959 je strojové učení způsob, jak dostat počítače do práce bez programování explicitně. Obecně se jedná o proces doladění učení, který postupně zlepšuje původní náhodný systém.

To znamená, že cílem je vytvořit umělou inteligenci, která bude schopna najít správné řešení ze špatného systému doladit parametry modelu. K tomu se v algoritmu strojového učení používá mnoho různých přístupů.



O projektu

Algoritmus strojového učení



Konkrétně v tomto projektu je základní přístup k strojovému učení (machine learning algorithm, ML) založen na neuroevoluci.

V této podobě strojového učení používají evoluční algoritmy, jako je například genetický algoritmus (genetic algorithm, GA), pro učení umělých neuronových sítí (artificial neuronové networks, ANN).

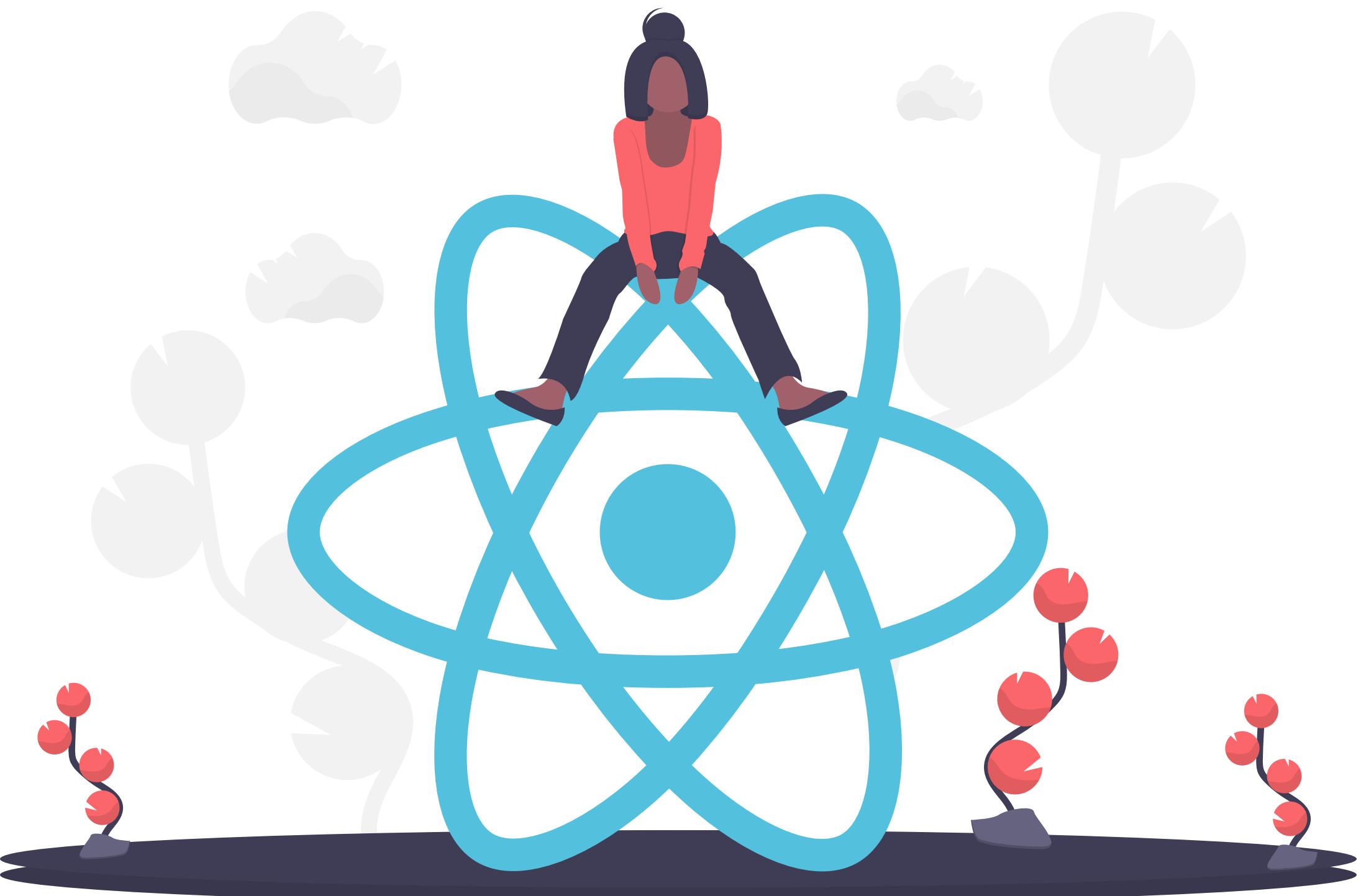
To znamená, že v našem případě lze říci, že $ML = GA + ANN$.

Umělá neuronová síť

Propojení sítě a strojového učení

Umělá neuronová síť je podmnožinou algoritmu strojového učení. Základem je struktura a funkce biologických neuronových sítí. Tyto sítě jsou vytvořeny z mnoha neuronů, které navzájem vysílají signály.

To znamená, že k vytvoření umělého mozku musíme stimulovat neurony a spojit je tak, aby vytvořily neuronovou síť.

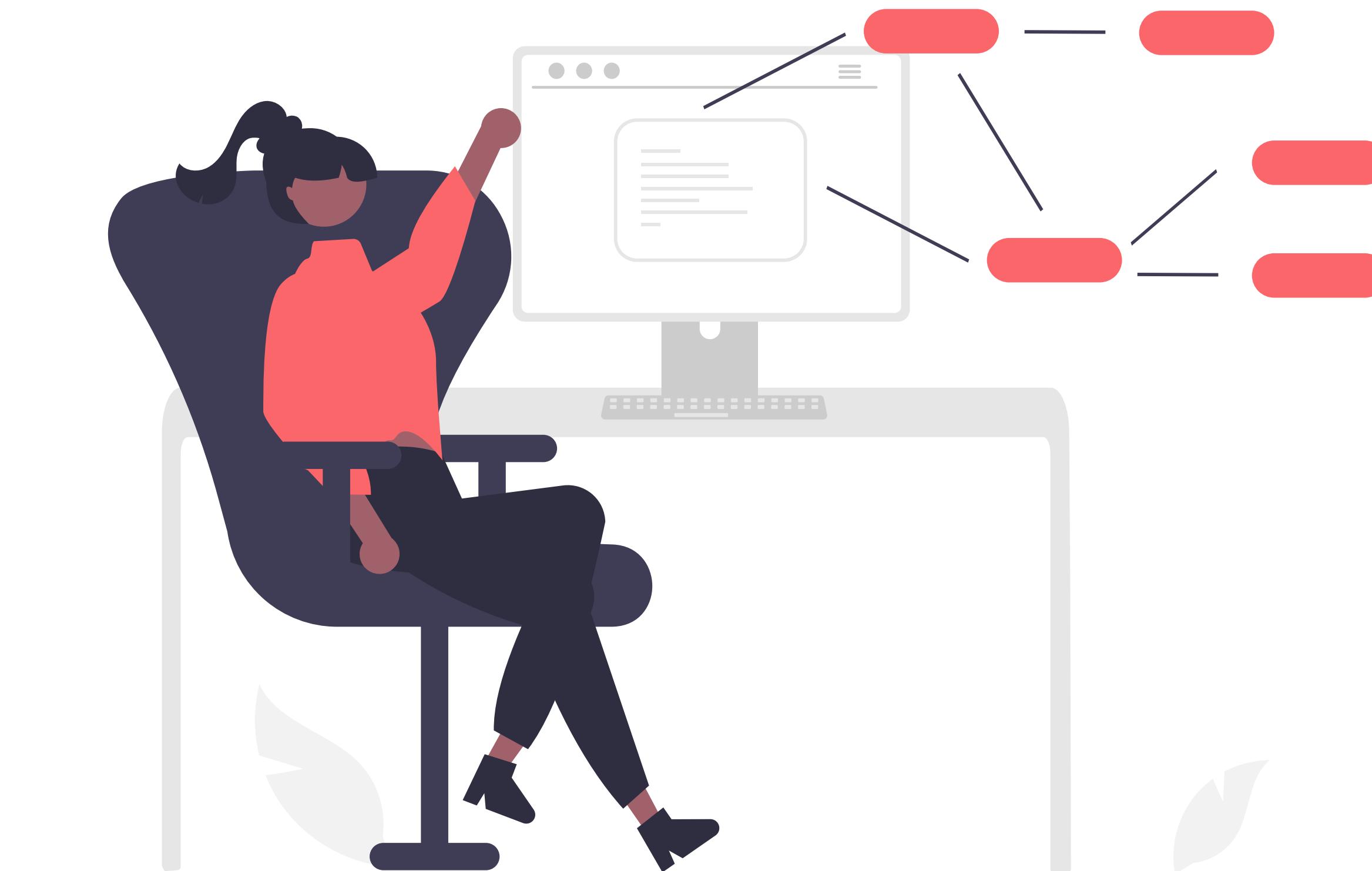


Umělá neuronová síť

Koncept vrstev

Standardní umělá neuronová síť se skládá z vstupní vrstvy, jedné nebo více skrytých vrstev a výstupní vrstvy. V každé vrstvě je několik neuronů.

Vstupní a výstupní neurony jsou připojeny přímo do vnějšího prostředí. Skryté neurony se mezi nimi spojují.



Umělá neuronová síť

Vrstvy sítě

V tomto projektu má každý objekt (pták) vlastní neuronovou síť používanou jako mozek AI, který prochází hrou.

Skládá se z následujících tří vrstev:

1

Vrstva vstupních dat se třemi neurony představuje to, co pták vidí:

- Horizontální vzdálenost mezi ptákem a nejbližší sadou trubek.
- Vertikální vzdálenost mezi ptákem a spodní trubkou.
- Vertikální vzdálenost mezi ptákem a horní trubkou.

2

Skrytá vrstva s dvaceti neurony.

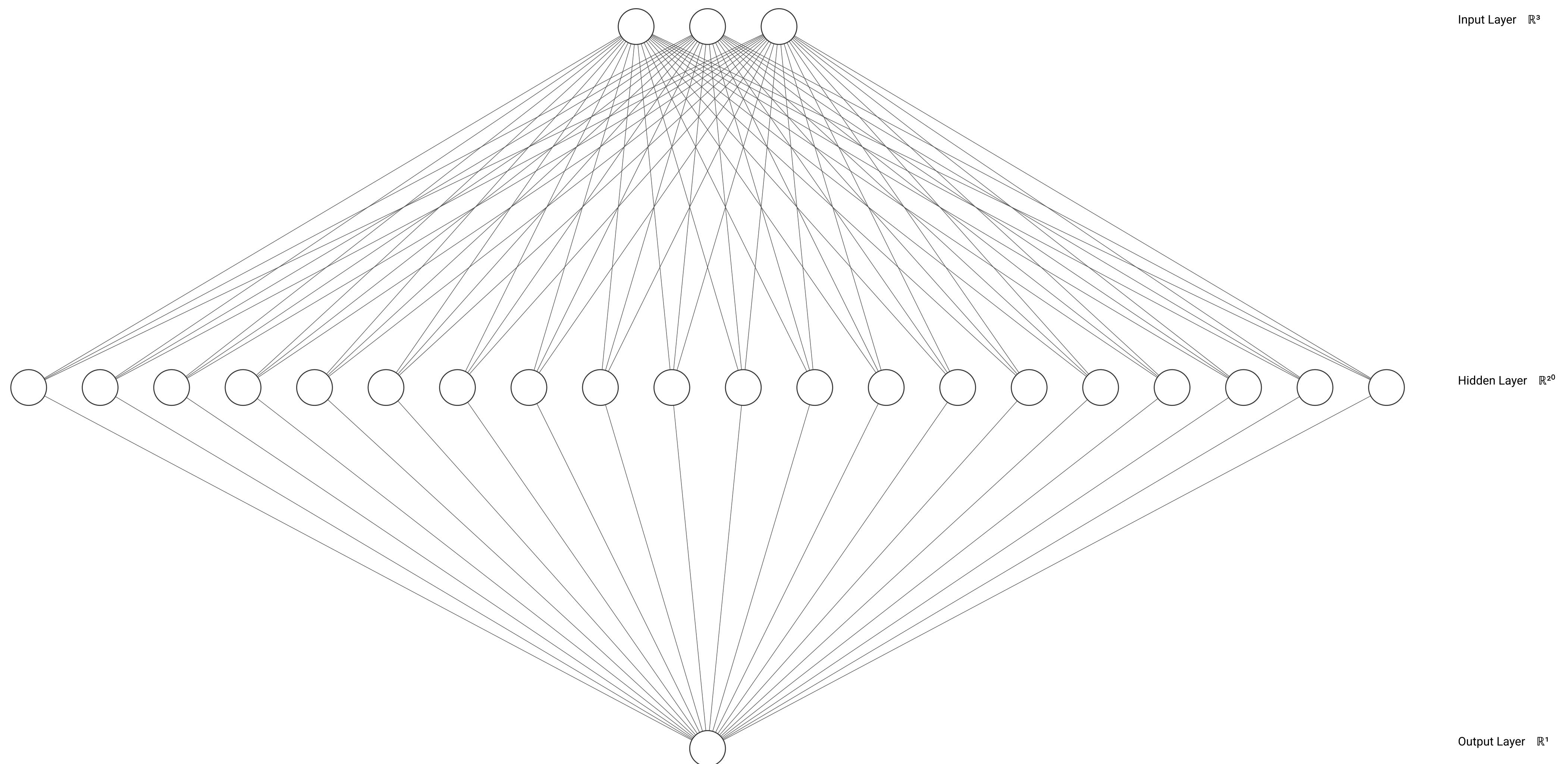
3

Výstupní vrstva s jedním neuronem, která vytváří akci -

- Je-li výstup 0,5, pak pták dělá mávnutí křídlem , jinak nedělat nic.

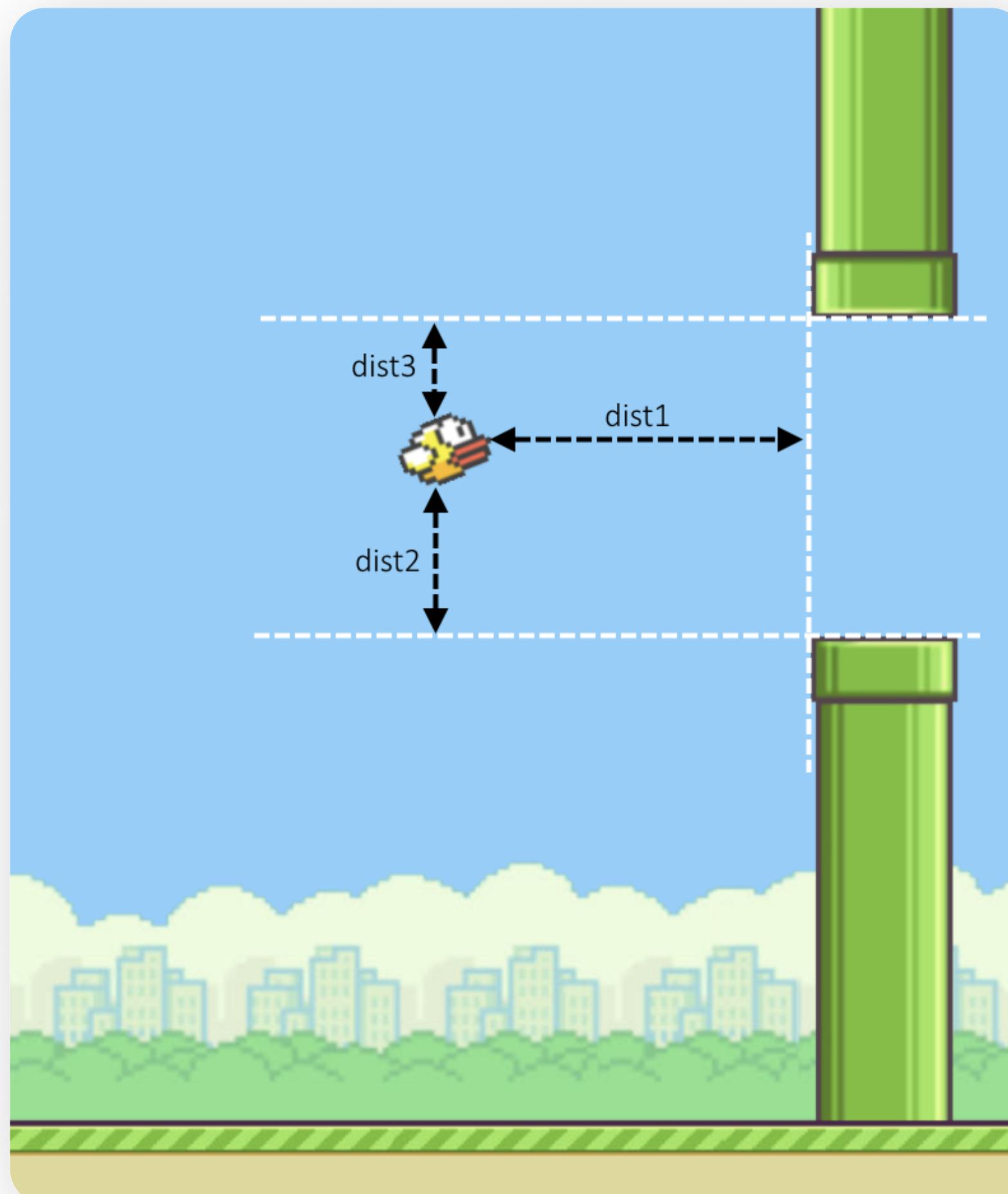
Umělá neuronová síť

Architektura sítě



Umělá neuronová síť

Vrstva #1



Pták vidí před sebou **tři parametry**, na základě kterých se může orientovat v dostupném prostoru.

Tyto tři parametry označíme jako vstupní neurony pro naši síť.

- Horizontální vzdálenost mezi ptákem a nejbližší sadou trubek.
- Vertikální vzdálenost mezi ptákem a spodní trubkou.
- Vertikální vzdálenost mezi ptákem a horní trubkou.

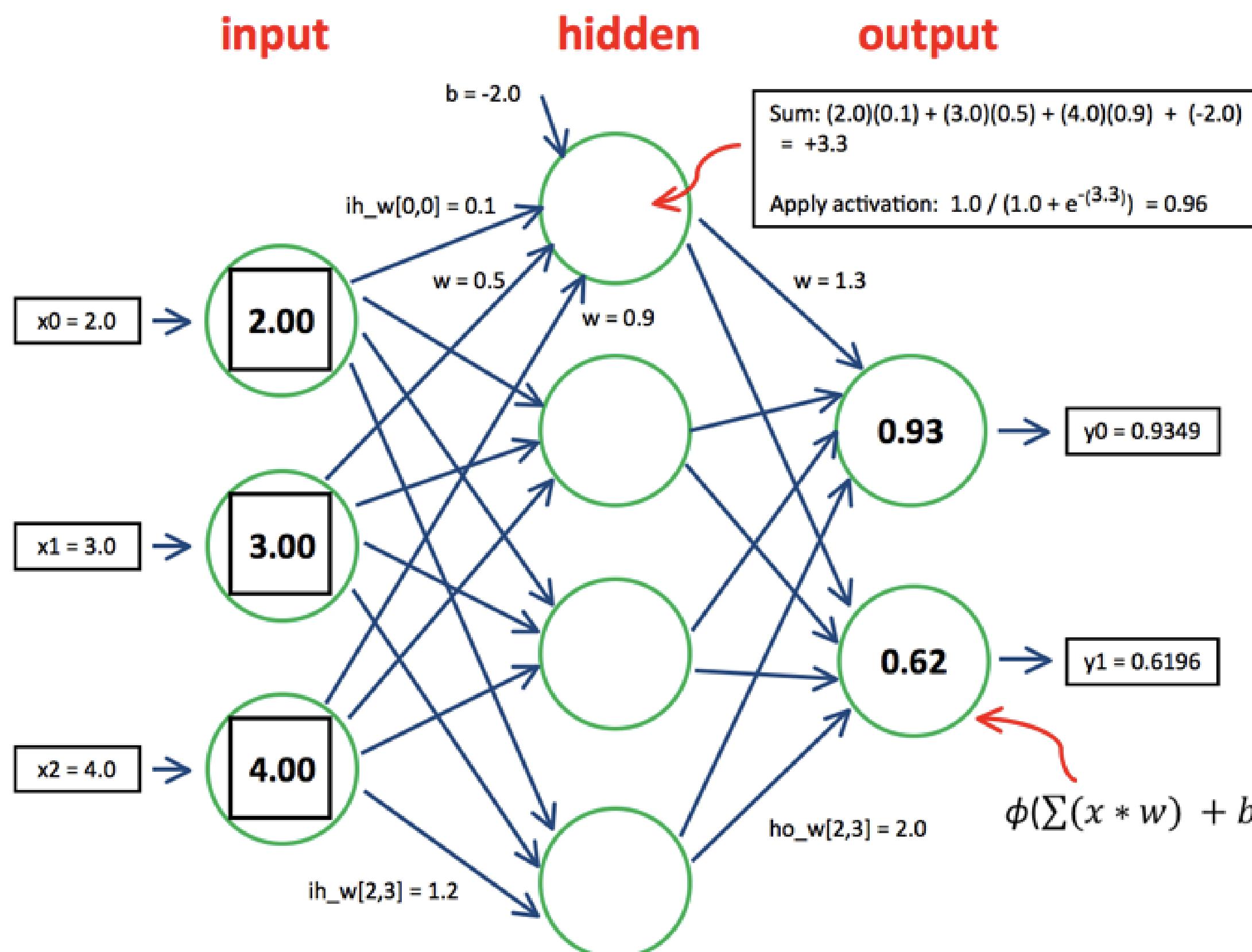
Dále přenášíme získaná data na druhou skrytou vrstvu.

Umělá neuronová síť

Vrstva #2

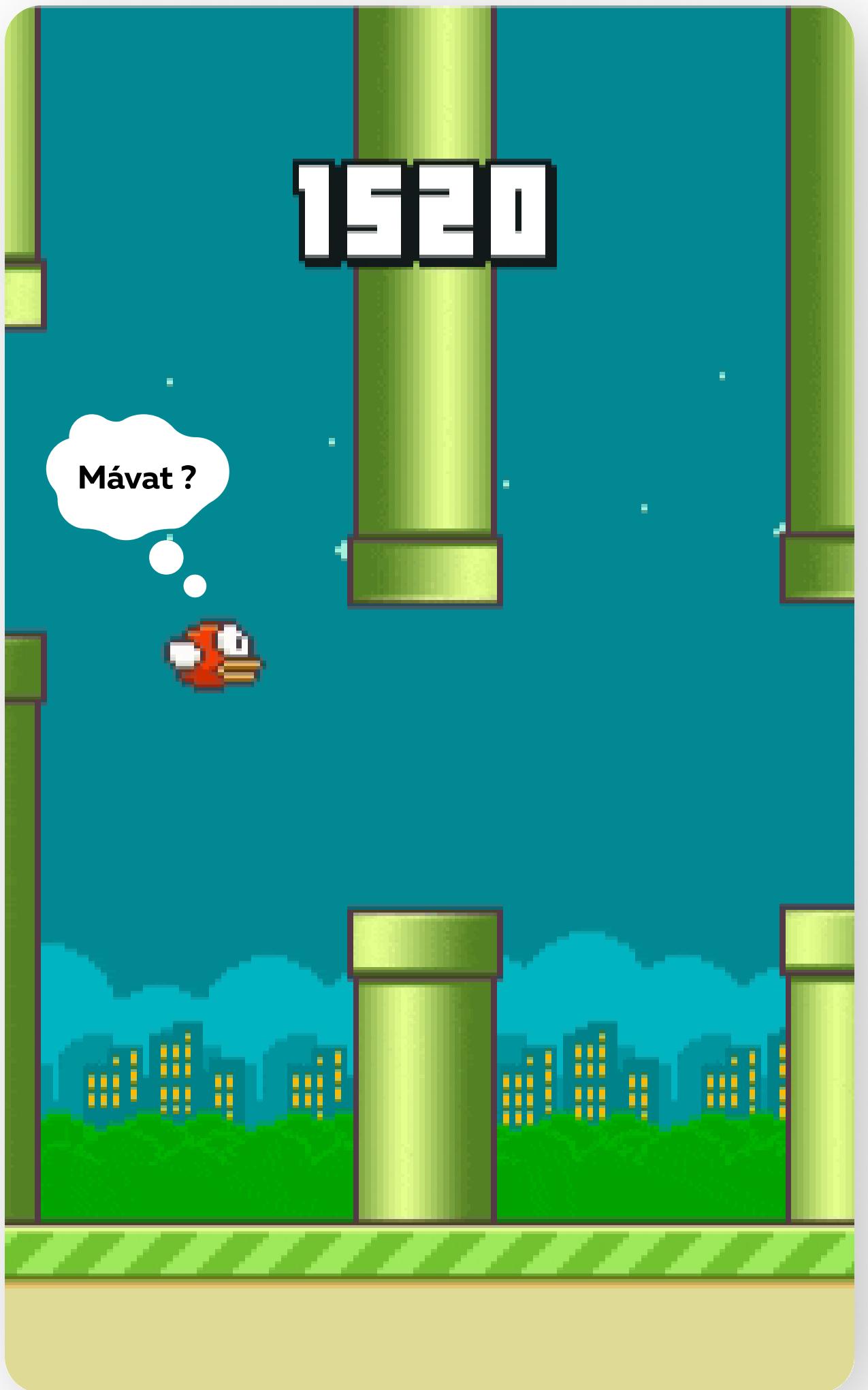
Každé připojení v síti má váhu a každý neuron má hodnotu zkreslení.

Níže uvedený graf například ukazuje, co se děje v neuronové síti.



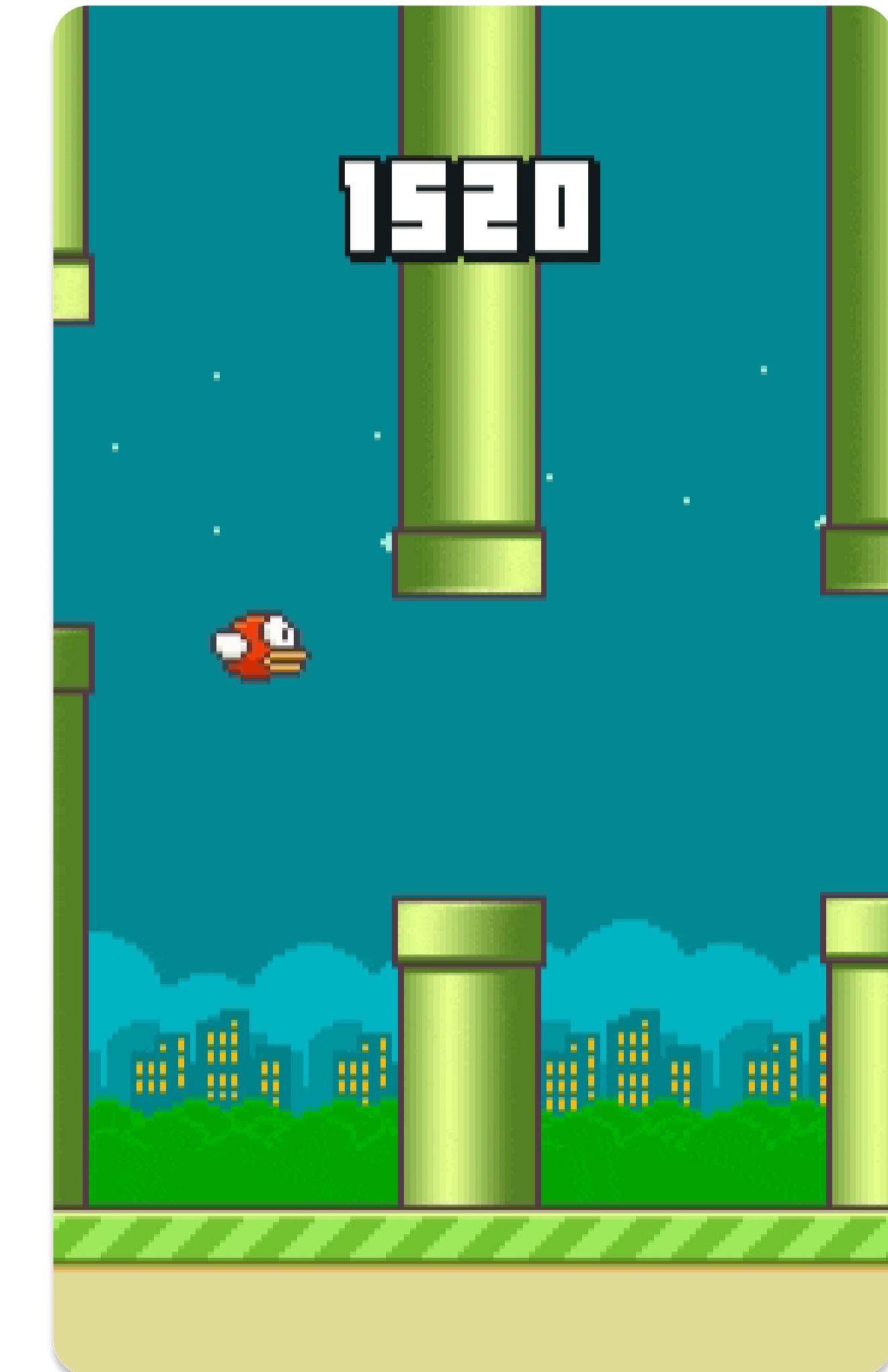
Umělá neuronová síť

Vrstva #3



Konečným krokem je sběr dat z druhé vrstvy shrnutí výsledků, po kterých se pták rozhodne, zda mávnout křídly nebo ne.

$$f(dist_1, dist_2, dist_3) = \begin{cases} > 0.5 \Rightarrow jump() \\ \leq 0.5 \Rightarrow do_nothing \end{cases}$$



Genetický algoritmus

Pochopení algoritmu

Když jsme diskutovali o algoritmu strojového učení, říkali jsme, že pro učení a zdokonalování neuronových sítí se používá genetický algoritmus.

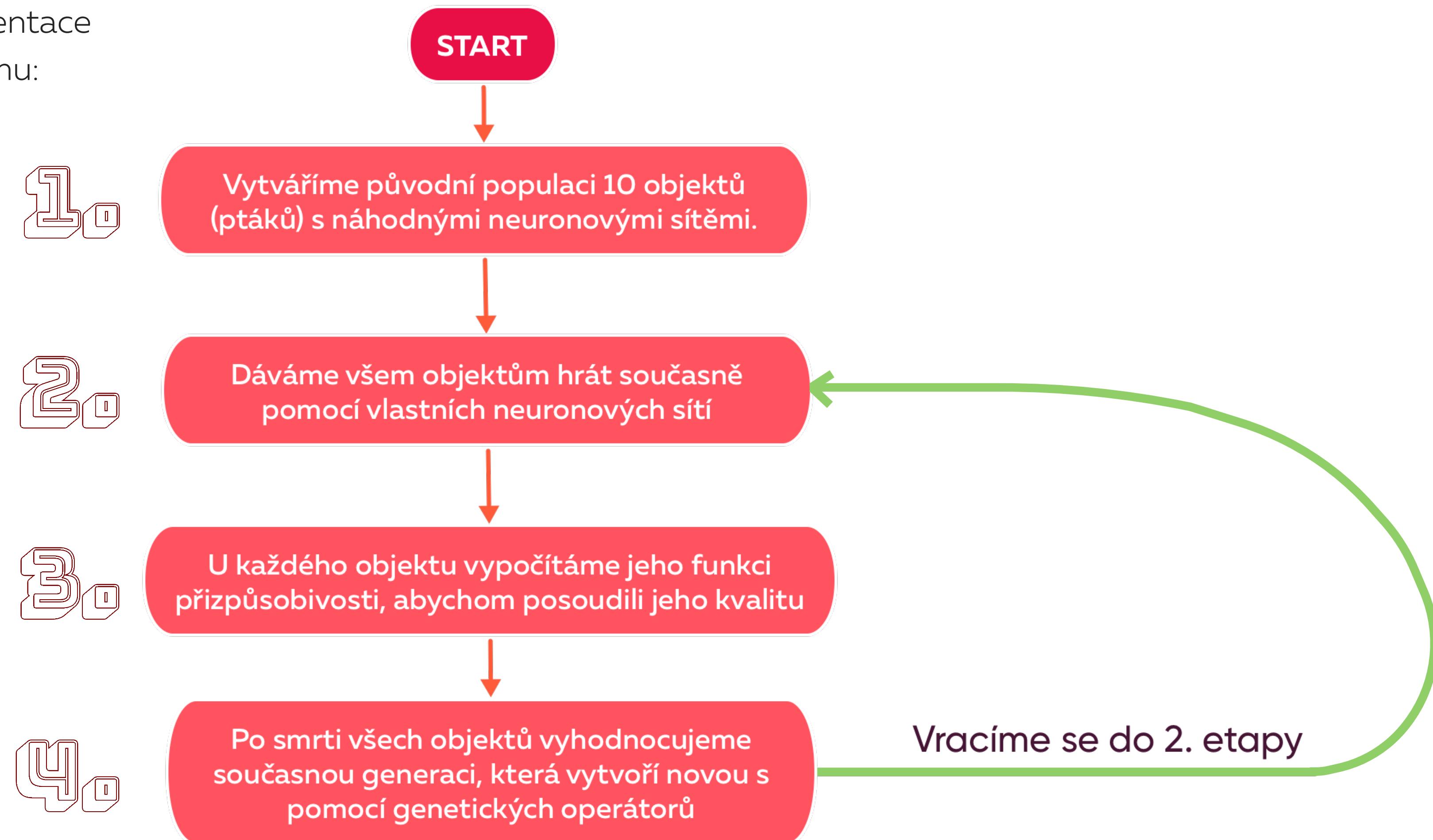
Genetický algoritmus je vyhledávací optimalizační technika, která kopíruje přirozený výběr a genetiku. Používá stejnou kombinaci výběru, crossingoveru a mutací ke změně původní náhodné populace.



Genetický algoritmus

Fáze realizace

Zde jsou hlavní fáze implementace našeho genetického algoritmu:



Funkce přizpůsobení

Její role a využití

Kromě genetického algoritmu (fáze 3) se zde podíváme na další funkce přizpůsobení – co to je a jak to určit.

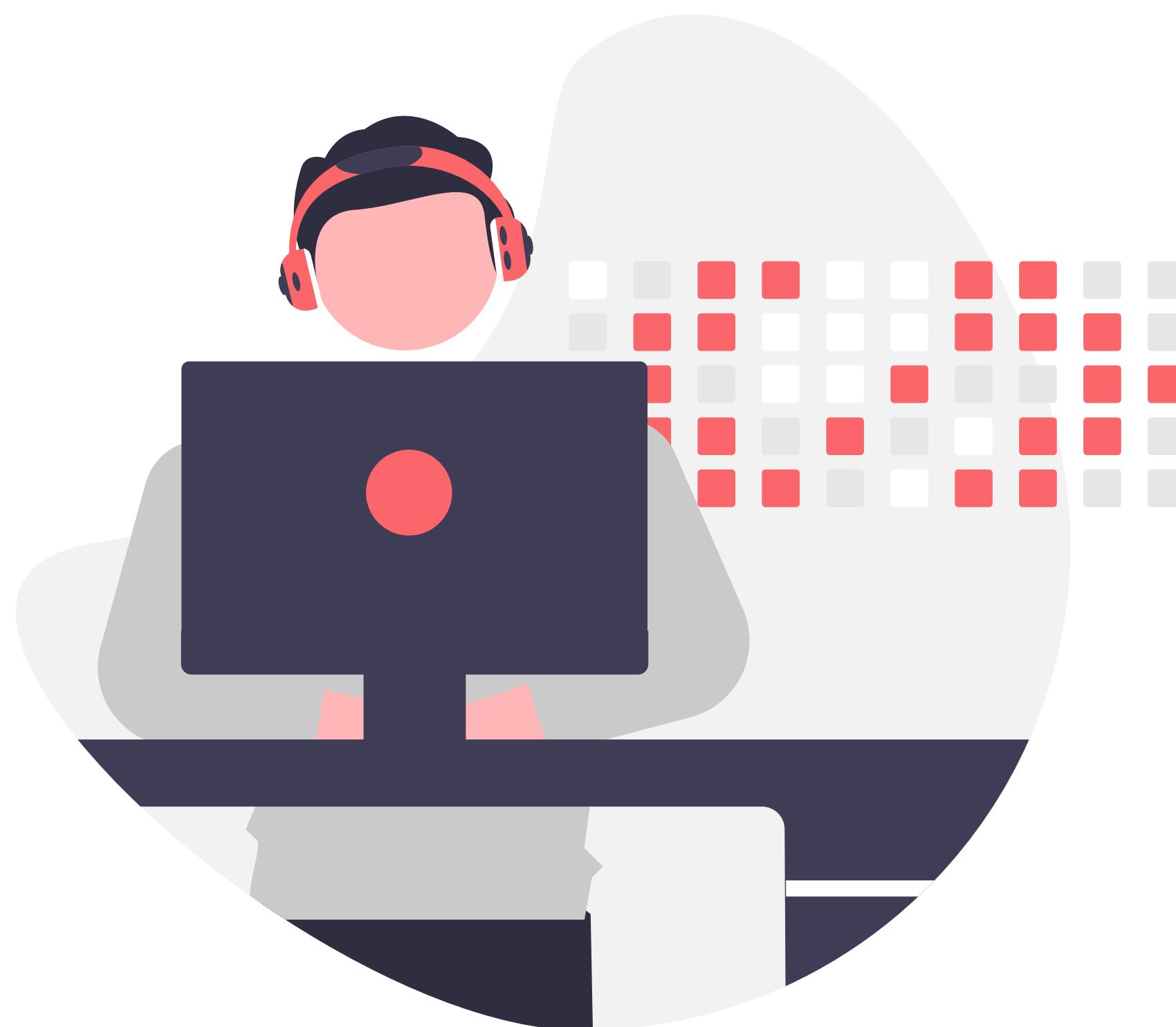
Protože chceme, aby se populace vyvinula z nejlepších objektů, musíme určit funkci přizpůsobení.

Obecně platí, že funkce přizpůsobení je metrika, která měří kvalitu objektu. Pokud budeme mít metriku kvality každého ptáka, budeme moci vybrat nevhodnější objekty a použít je k opětovnému vytvoření nové generace.



Funkce přizpůsobení

Její role a využití



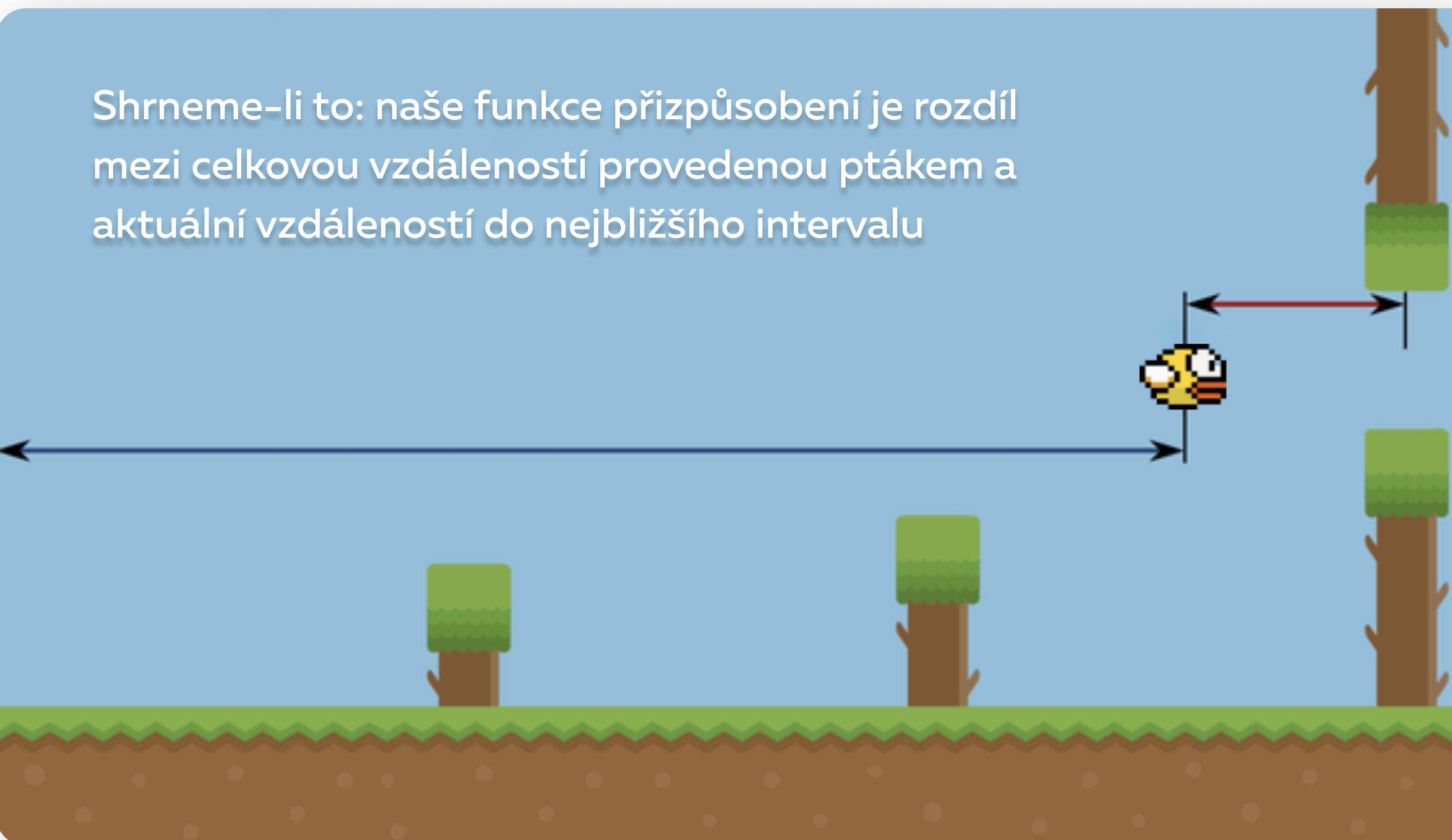
V tomto projektu odměňujeme ptáka v přímé závislosti na vzdálenosti. Navíc ji trestáme podle aktuální vzdálenosti do nejbližší vzdálenosti. Tímto způsobem budeme schopni rozlišovat mezi ptáky, kteří letěli stejnou vzdálenost.

Shrneme-li to: naše funkce přizpůsobení je rozdíl mezi celkovou vzdáleností provedenou ptákem a aktuální vzdáleností do nejbližšího intervalu.

Funkce přizpůsobení

Co tím myslíme ?

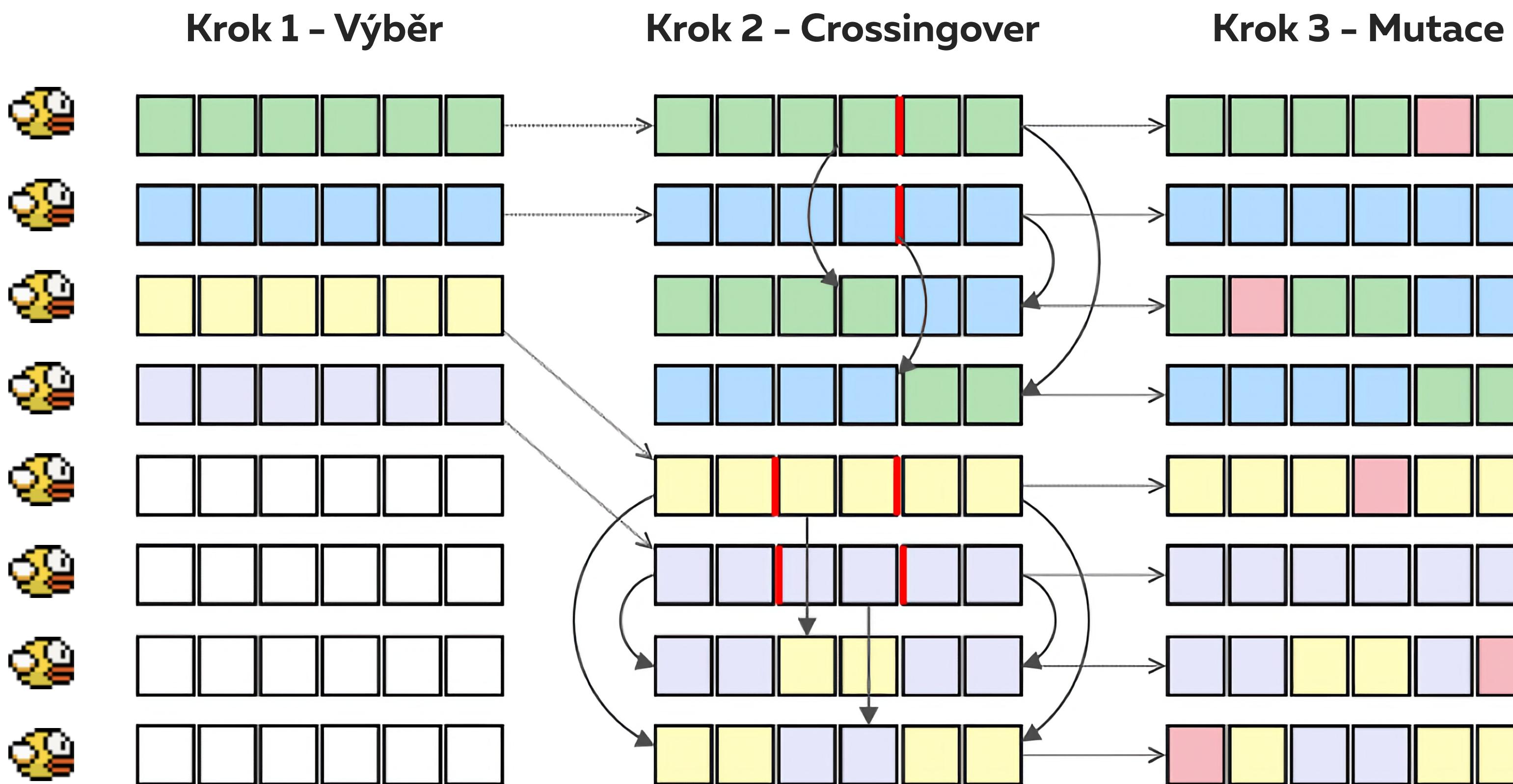
V tomto projektu odměňujeme ptáka v přímé závislosti na vzdálenosti. Navíc ji trestáme podle aktuální vzdálenosti do nejbližší vzdálenosti. Tímto způsobem budeme schopni rozlišovat mezi ptáky, kteří letěli stejnou vzdálenost.



Strategie nahrazení

Funkce náhrady v programu

Kromě genetického algoritmu (fáze 4) jsou zde fáze aplikace přirozené evoluce pro umírající generaci. V podstatě přežívají lepší objekty a jejich potomci nahrazují nejhorší objekty následovně:



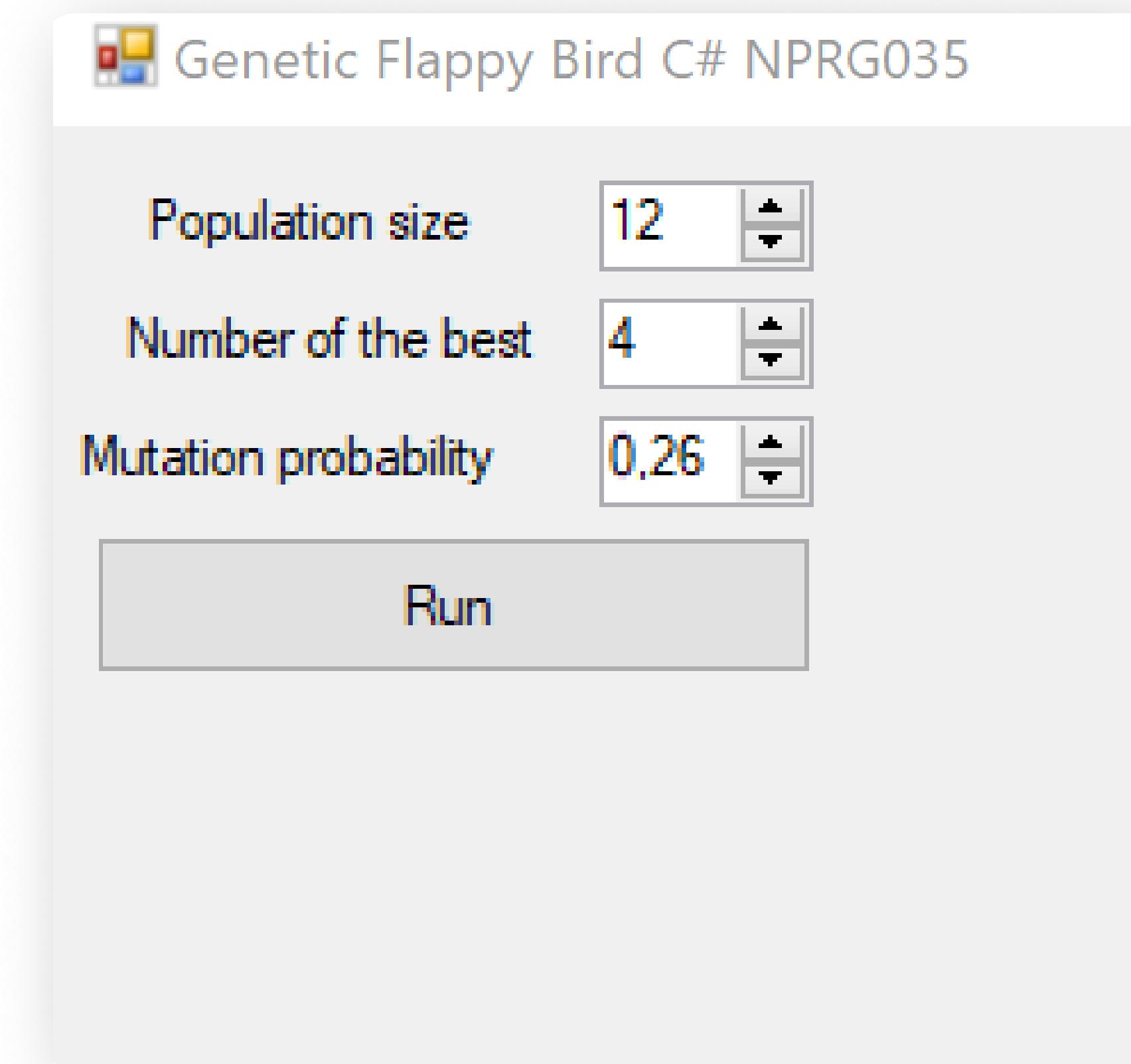
Testování programu

Přecházíme z teorie do praxe

Přejdeme do projektu a spustíme aplikaci.

Poté vidíme před sebou vše potřebné pro nastavení funkce:

- Velikost populace
- Počet nejlepších
- Pravděpodobnost mutace



Testování programu

Přecházíme z teorie do praxe

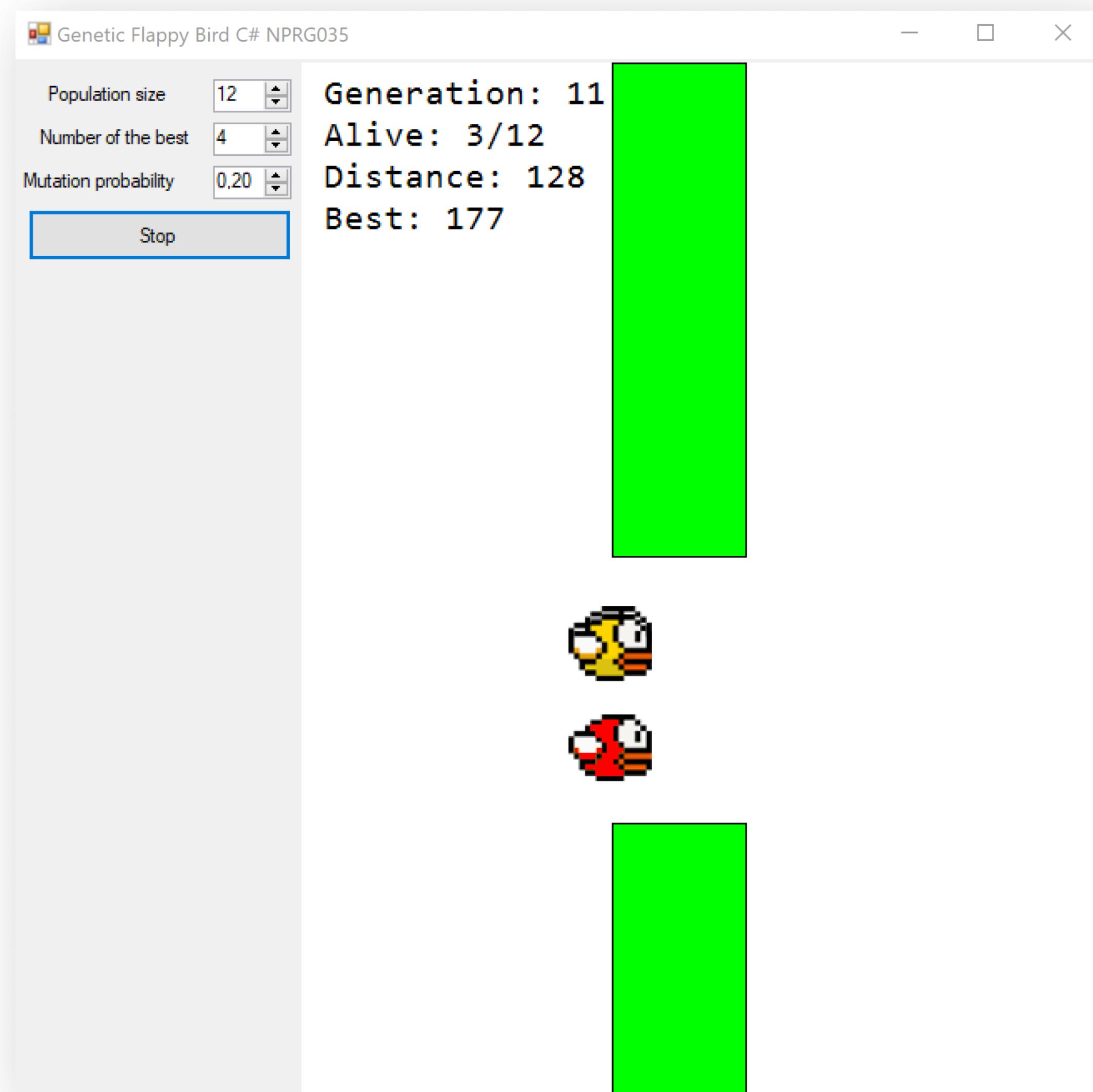
Zadáváme hodnoty, které potřebujeme.

A zmáčkneme start.

Začal proces učení ptáků.

Zpočátku uvidíme, že prvních pět až deset generací se neúspěšně pokusí projít první překážkou.

Poté si můžete všimnout, že jeden nebo dva ptáci již jistě procházejí prvními překážkami. A snaží se jít pokaždé dál.



Testování programu

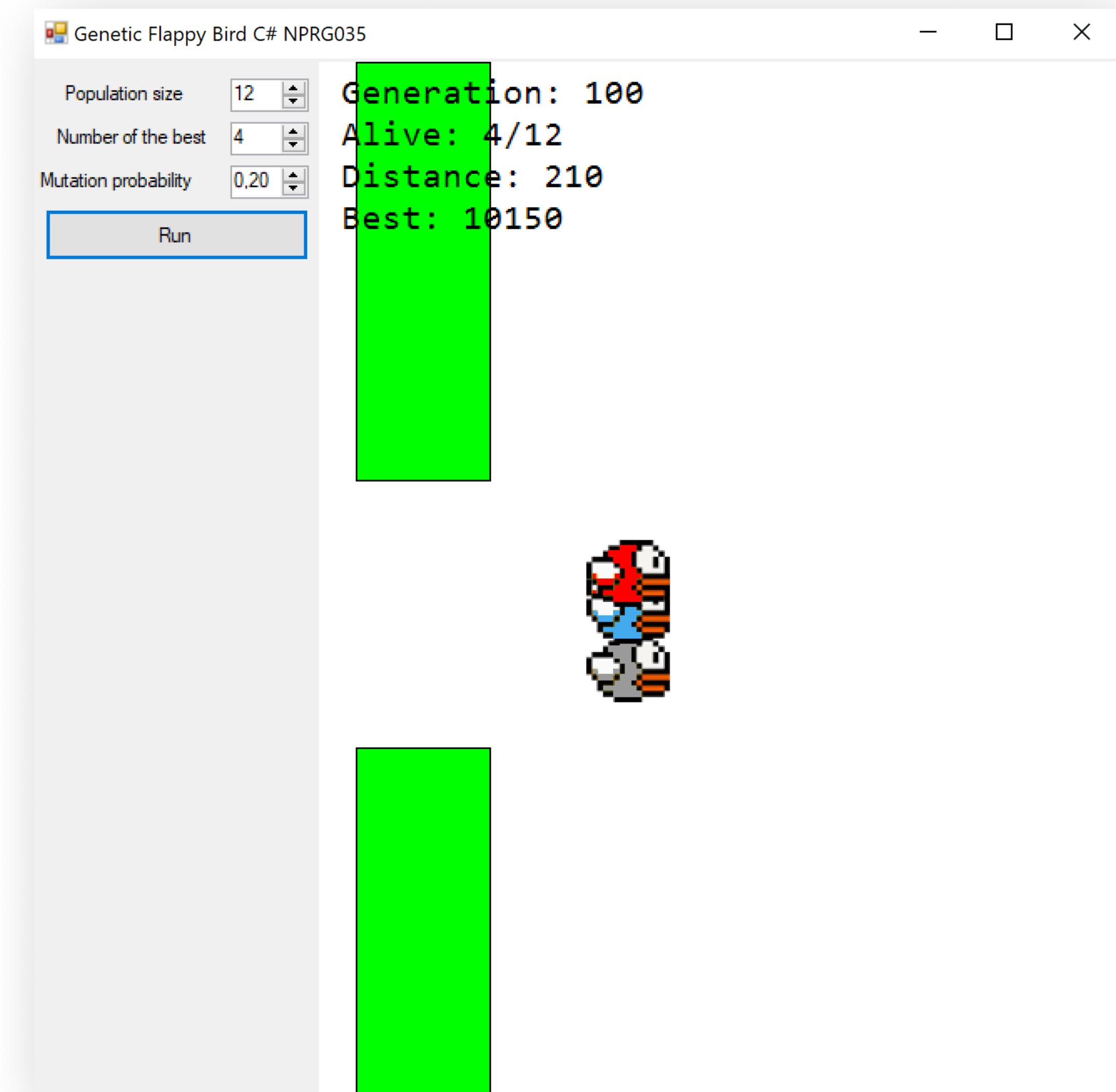
Přecházíme z teorie do praxe

S každou generací se ptáci stávají chytřejšími a procházejí stále větší vzdáleností než jejich předchůdci.

Nakonec se jeden z ptáků podaří dosáhnout maxima svých možností (v určitém okamžiku, čistě fyzicky, pták nemůže mít čas projít mezi trubkami kvůli extrémně vysoké rychlosti).

Tak vysoko můžeme počítat vzdálenost 10 000.

A s každou novou generací budeme moci vidět, že na maximum je již 4-6 ptáků.
Nakonec na maximum dosáhne většina dostupných ptáků.



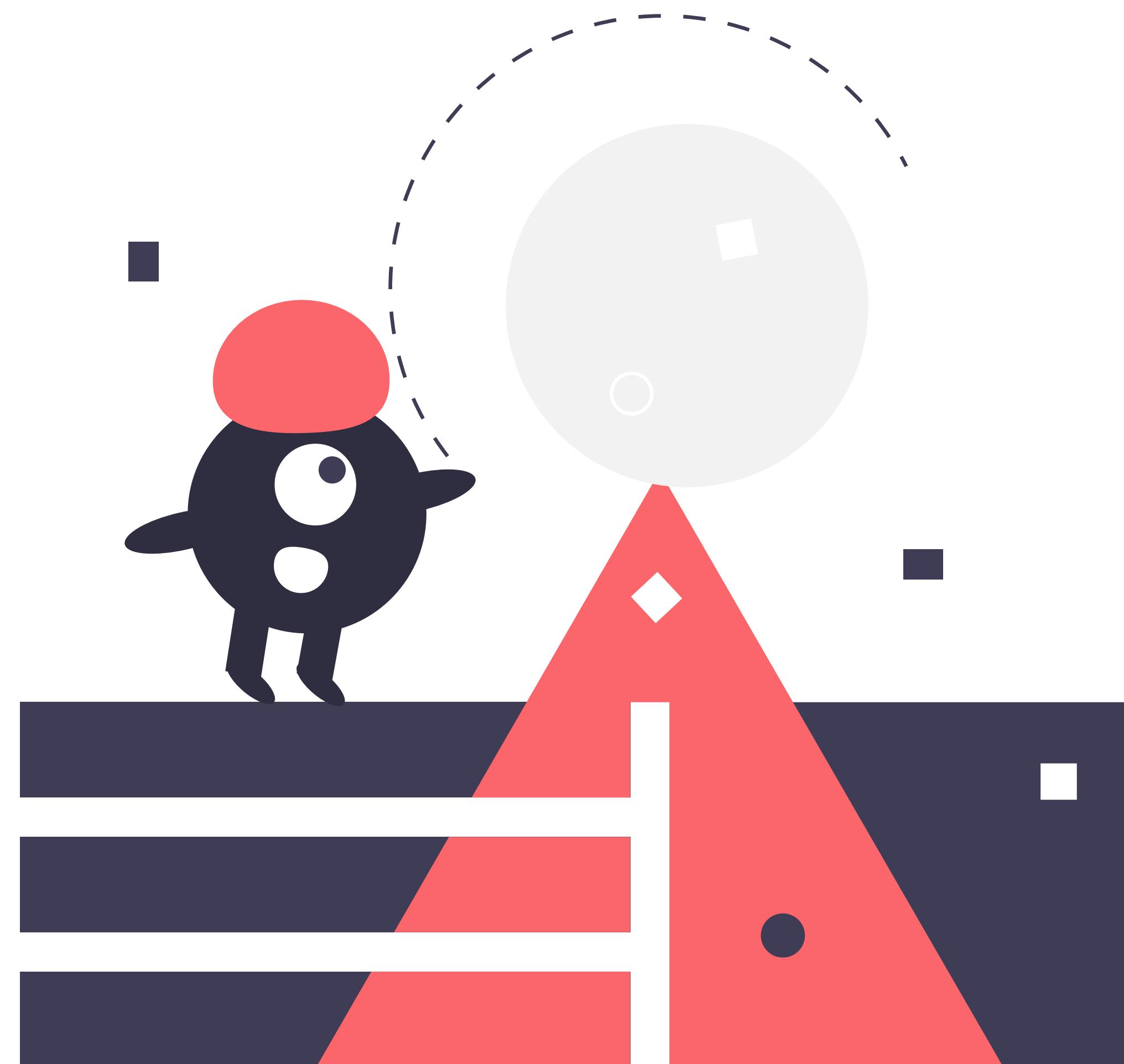
Závěr

Shrnutí práce

V tomto projektu jsme úspěšně implementovali AI robota pro výuku hry Flappy Bird.

V důsledku několika iterací můžeme získat téměř nezranitelného hráče. K dosažení tohoto cíle jsme použili dva přístupy k algoritmům strojového učení: umělé neuronové sítě a genetický algoritmus.

Tento přístup lze aplikovat i na vývoj v jiných hrách.



Použité zdroje

Upřímně děkuji každému autorovi za jeho práci!



- <https://undraw.co>
- <https://habr.com/ru/post/336612/>
- <https://codingvision.net/c-making-a-neural-network-that-plays-flappy-bird>
- <https://towardsdatascience.com/neural-network-plays-flappy-bird-e585b1e49d97>
- <https://medium.com/analytics-vidhya/how-i-built-an-ai-to-play-flappy-bird-81b672b66521>
- <https://neurohive.io/ru/osnovy-data-science/osnovy-nejronnyh-setej-algoritmy-obuchenie-funkcii-aktivacii-i-poteri/>