

# Application of Genetic Algorithm

Yudi Umar

# Introduction

---

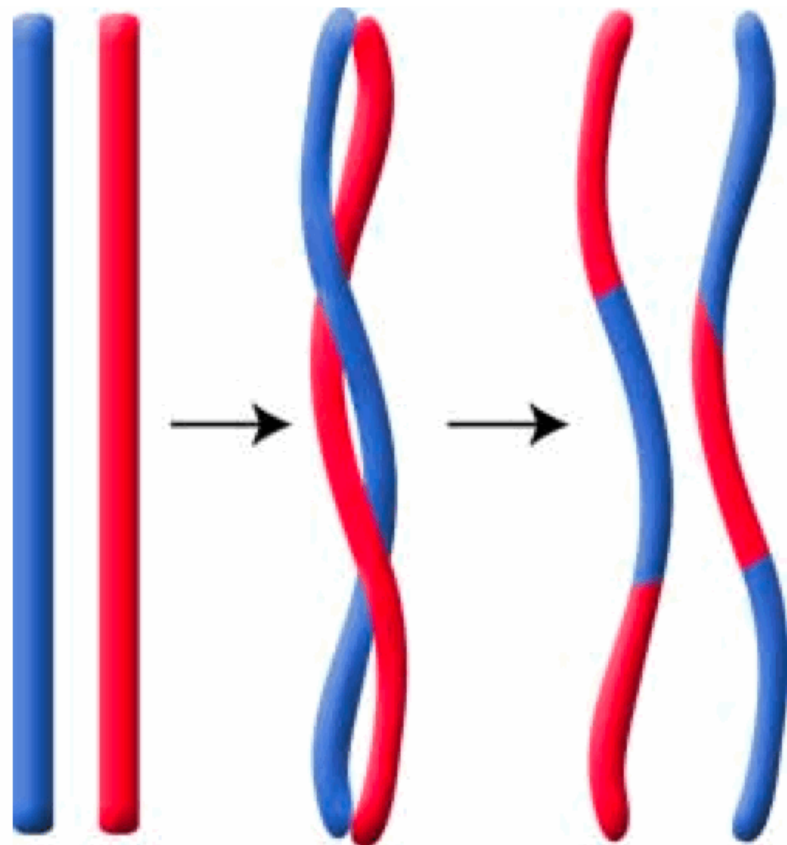
- ▶ Genetic Algorithm (GA) digagaskan oleh John Henry Holland pada 1970an
- ▶ Saat sekarang sering digunakan untuk permasalahan optimisasi



# Introduction

---

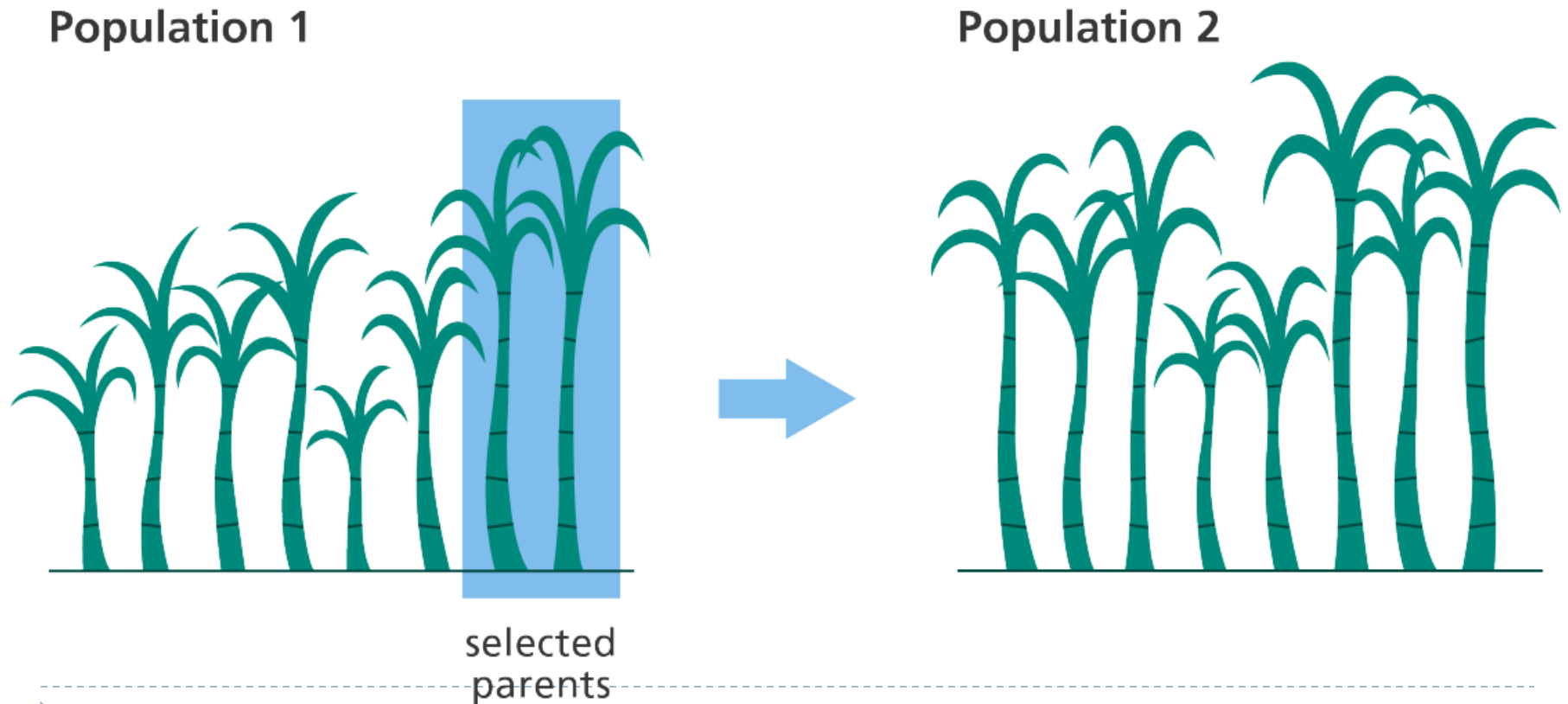
- ▶ Dalam biologi Genetik, chromosomes bertukar DNA pada fase meiosis



# Introduction

---

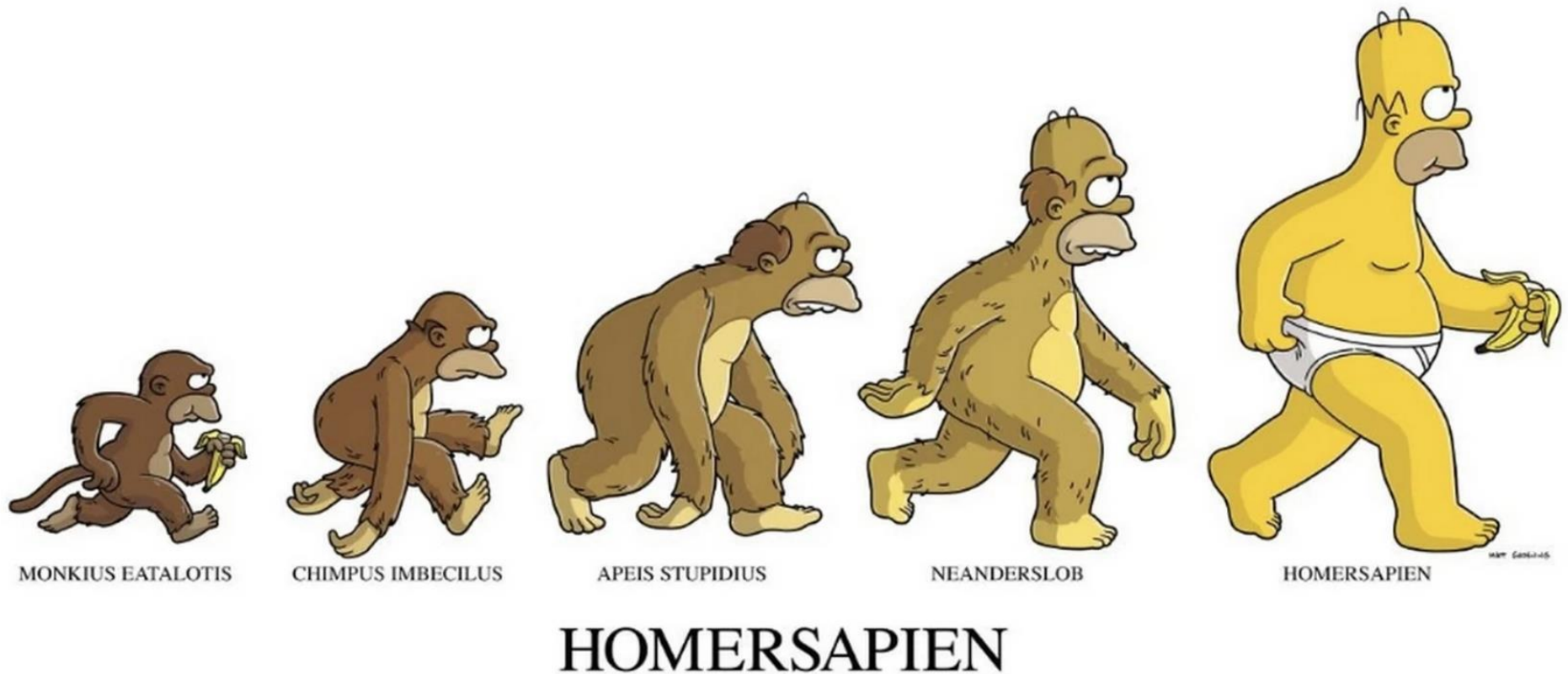
- ▶ GA terinspirasi dari proses seleksi buatan



# Introduction

---

- ▶ GA bagian dari Evolutionary Algorithm



# How it works

---

- ▶ Initialization
- ▶ Selection
- ▶ Offsprings : Genetic Operators
  - ▶ Cross Over
  - ▶ Mutation
- ▶ Go to Selection, until termination



# How it works - Initialization

---

- ▶ Individu = Chromosome
- ▶ Chromosome => representasi kebiasaan (DNA)
- ▶ Mulai dengan populasi (kumpulan individu) yang acak



# How it works - Selection

---

- ▶ Ada penilaian terhadap suatu individu (tinggi, manis, besar, dll) => Fitness function
- ▶ Selectif memilih yang baik untuk disilangkan
- ▶ Yang kurang baik akan ditinggalkan





# How it works - Offsprings

---

- ▶ **Cross Over**

- ▶ Menyilangkan antara 2 buah individu untuk dijadikan sebagai induk, mengambil sebagian kebiasaan (DNA)

- ▶ **Mutation**

- ▶ Ada sedikit perubahan sifat



# How it works - Repeat

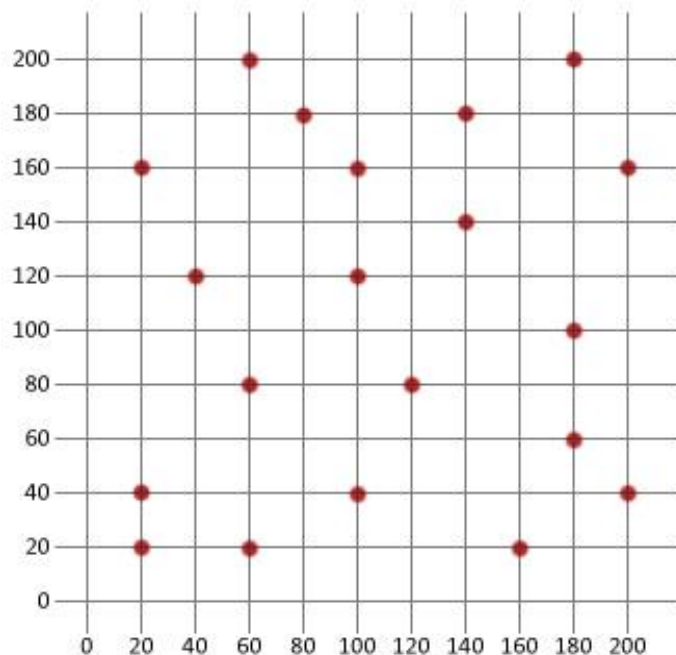
---

- ▶ Ulangi terus menerus, hingga mendapatkan generasi yang baik



# Sample

- ▶ Contoh kasus penerapan GA pada Travelling Salesman Problem
- ▶ Terdapat N kota, si Salesman berkunjung ke semua kota wajib 1x dan dengan total jarak tempuh terkecil



Titik merah = Kota

Dinomori dari 1 s/d N

Jarak 2 Kota dihitung dengan jarak Euclidean

# Sample - Initialization

---

- ▶ Representasi kebiasaan dari Salesman (Chromosome) adalah sekumpulan angka 1 s/d N (DNA)
- ▶ Contoh  $N = 4$ .
  - ▶ 1 4 3 2  $\Rightarrow$  Kota 1  $\rightarrow$  Kota 4  $\rightarrow$  Kota 3  $\rightarrow$  Kota 2
- ▶ Populasi  $P$  dibuat dengan individu acak
  - ▶ For  $i = 1 \dots P$ 
    - ▶ Individu[i] = random\_shuffle([1..N])



# Sample - Selection

---

- ▶ Goal dari TSP adalah total jarak tempuh terkecil
- ▶ Fitness function dari chromosome mendapatkan jarak tempuh terkecil. Makin kecil makin baik
- ▶ Hitung total jarak Euclidean dari kota yang bersinggungan
- ▶ Contoh  $N = 4$ .
  - ▶  $1\ 4\ 3\ 2 \Rightarrow \text{Kota } 1 \rightarrow \text{Kota } 4 \rightarrow \text{Kota } 3 \rightarrow \text{Kota } 2$
  - ▶  $\text{Fitness function} = \text{Jarak}(1, 4) + \text{Jarak}(4, 3) + \text{Jarak}(3, 2)$
- ▶ Dari  $P$  populasi dipilih sebanyak  $P_s$  sebagai individu unggulan



# Sample – Offsprings

---

## ▶ Cross Over (PMX)

▶ P1            3 4 8 **2 7 1** 6 5

▶ P2            4 2 5 **1 6 8** 3 7

▶ Block merah bertukar, sisanya tidak konflik hitam

▶ O1            3 4 **2 1** **6 8** **7** 5

▶ O2            4 **8** 5 **2 7 1** 3 **6**

## ▶ Mutation

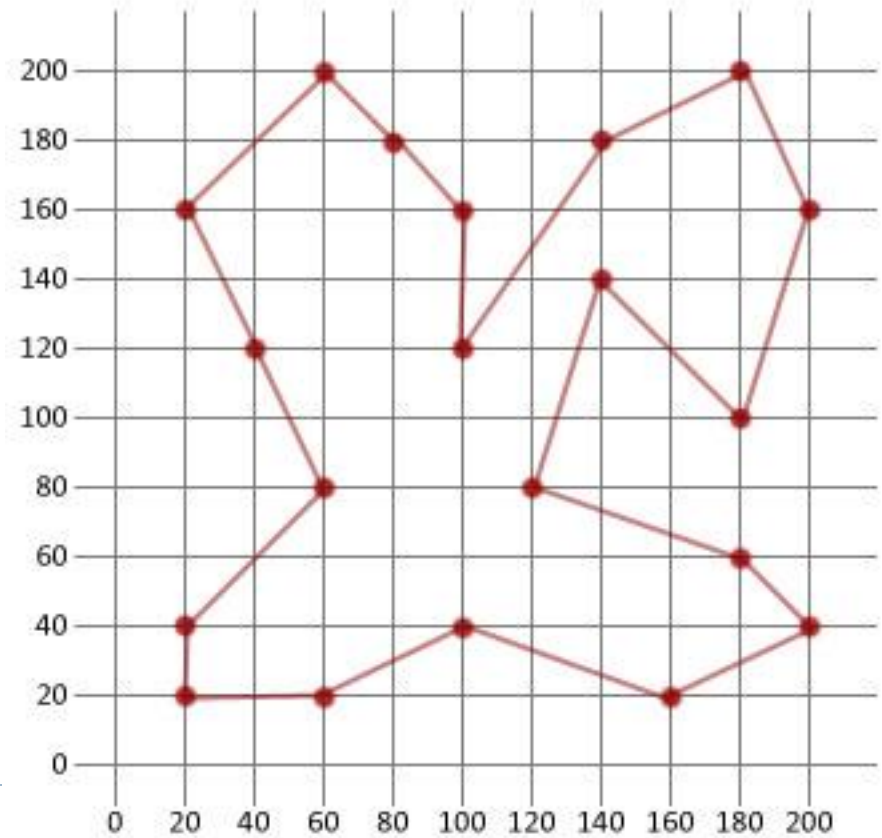
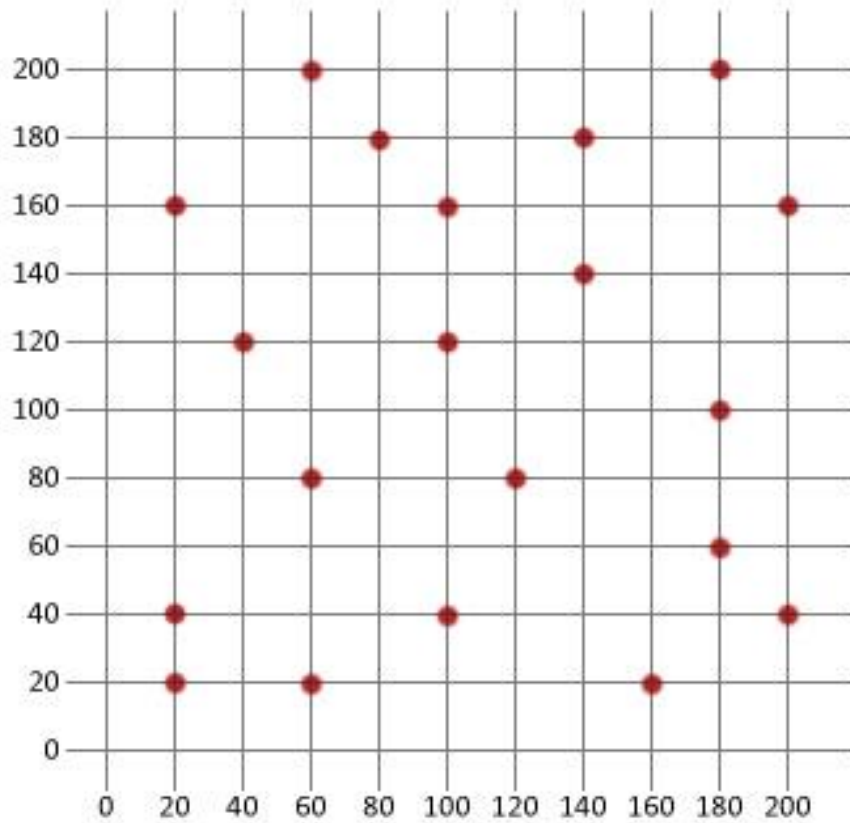
▶ Random swap

▶ O1            3 4 **2** 1 6 **8** 7 5

▶ O1<sub>new</sub>        3 4 **8** 1 6 **2** 7 5



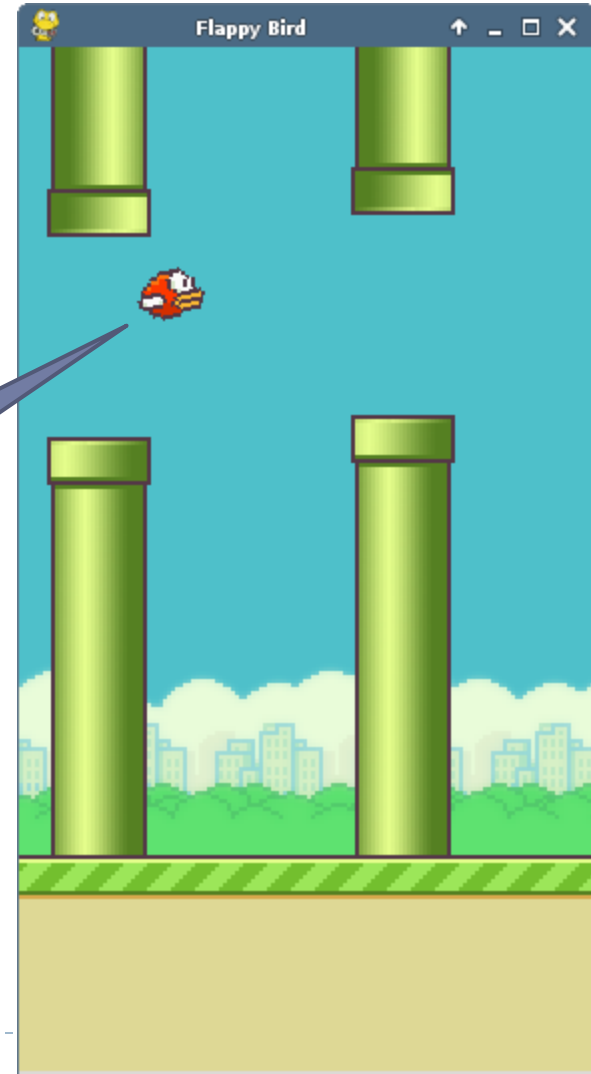
# Sample - Result



# Application

- ▶ Flappy Bird
- ▶ Flap tidak boleh mengenai pipa

EAAAAAKKKK

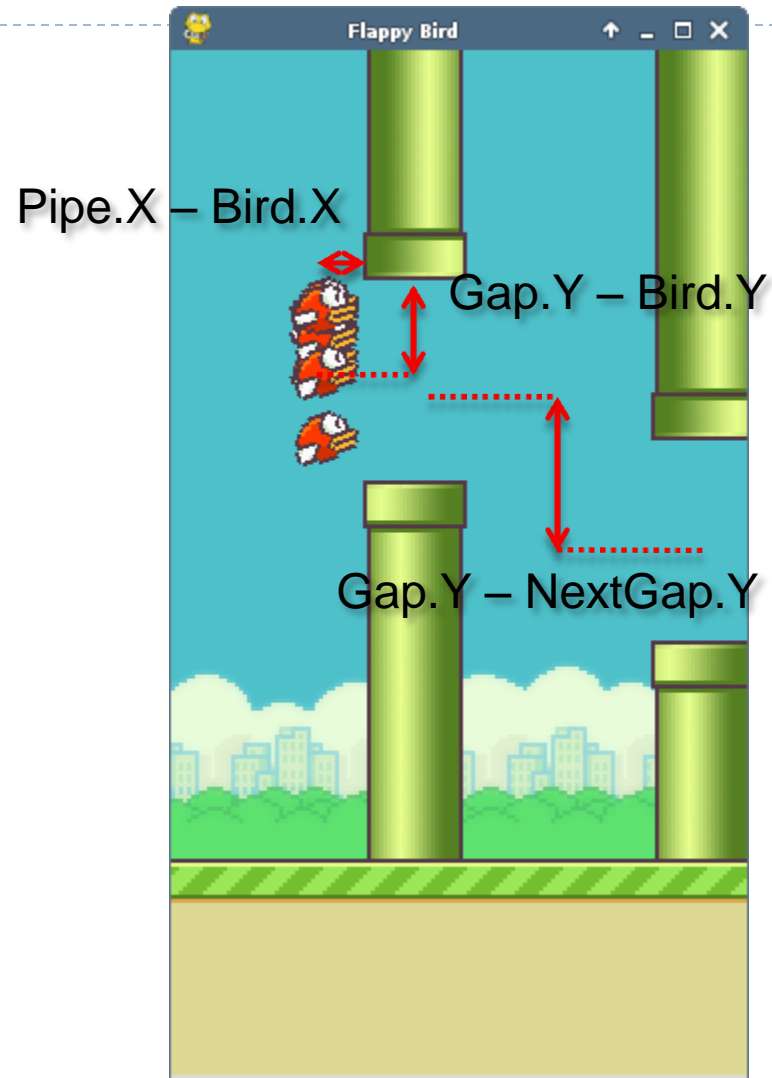




# Application - Initialization

- ▶ Chromosome : konstanta terhadap variabel berikut
  - ▶ Jarak Bird dengan Pipa ( $\text{Pipe.X} - \text{Bird.X}$ )
  - ▶ Selisih tinggi Bird dengan Center Gap Pipa ( $\text{Gap.Y} - \text{Bird.Y}$ )
  - ▶ Selisih Gap Pipa dengan Pipa berikutnya ( $\text{Gap.Y} - \text{NextGap.Y}$ )
- ▶ Populasi  $P = 20$





# Application - Selection

---

- ▶ Goal adalah si Bird bisa terus flap sampe lelah
- ▶ Function fitness Bird.X travel paling jauh + Bird.Y paling dekat dengan Gap.Y



# Application - Offsprings

---

## ▶ Heuristic

- ▶ Order by Score desc, bagi 3 kelas 30%, 40%, 30%
- ▶ 70% crossover dari 3 kelas
- ▶ 20% mutation dari offspring
- ▶ 10% parent + parent mutation

## ▶ Crossover

- ▶ Random swap konstanta

## ▶ Mutation

- ▶ Random + / - 0.01 s/d 0.1



# Application - Result

---



# Optimization Problem

---

