

EDUNEX ITB



## Modul 3: Beyond Classical Search

### Genetic Algorithm

KK IF – Teknik Informatika – STEI ITB

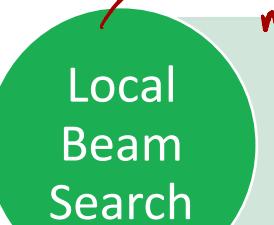
Inteligensi Buatan  
(Artificial Intelligence)



# Genetic Algorithm



- parallel instead of in sequence
- not independent: passing *useful information*



informasi successor  
dari setiap statenya  
masih di  
pertimbangkan.

chooses k  
successors  
at random

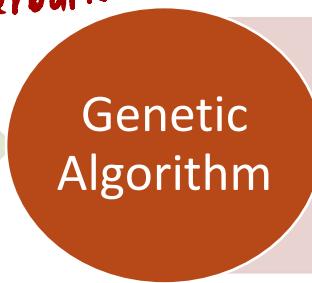
mulai dengan  
sejumlah k  
state.  
jd seolah² random  
restart tapi  
berjalannya  
parallel.

misal ada 4 state,  
casus 8 rata, successornya  
ada  $56 + 56 + 56 + 56$ ,  
diambil 4 terbaik



successor  
states:  
combining  
two parent  
states

↳ tidak harus lebih  
baik tetangganya,  
boleh lebih  
buruk dgn  
probabilitas  
tertentu.  
(ndak selalu  
memilih k  
best successor)



## Local Beam Search

→ masih ada kemungkinan  
stuck di local optima

function BEAM-SEARCH(*problem*,  $k$ ) returns a solution state

start with  $k$  randomly generated states

loop

generate all successors of all  $k$  states

if any of them is a solution then return it

else select the  $k$  best successors

begins with  $k$  randomly generated states.

all the successors of all  $k$  states are generated.

If any one is a goal, the algorithm halts. Otherwise, it selects the  $k$  best successors from the complete list and repeats.

In a local beam search, useful information is passed among the parallel search  $k$  threads

# Genetic Algorithm: Illustration (Russel & Norvig, 2010)



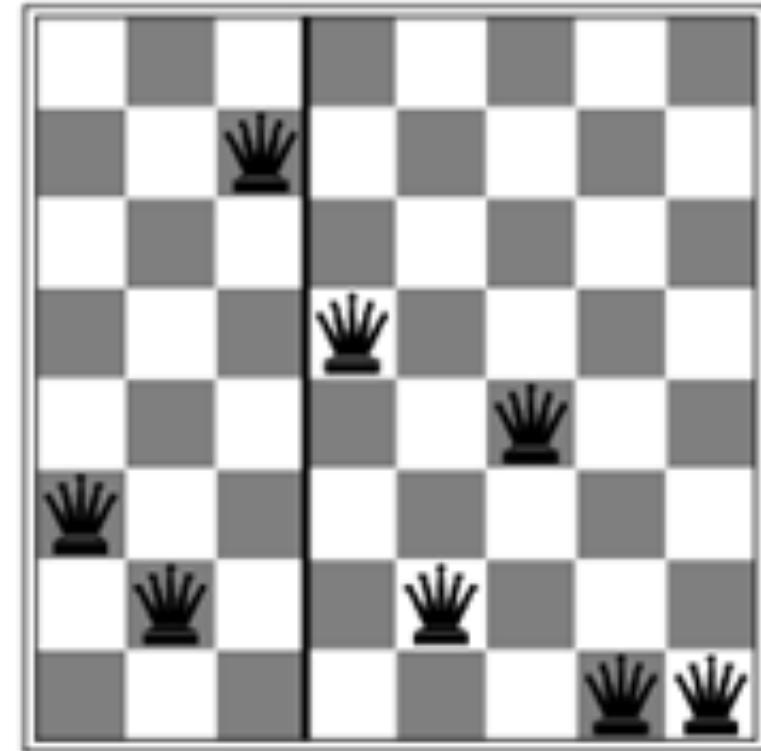
# State / Individual

A state is represented as a string over a finite alphabet  
(often a string of 0s and 1s)

One character ~ one variable

contoh representasi  
individu / state

32752411



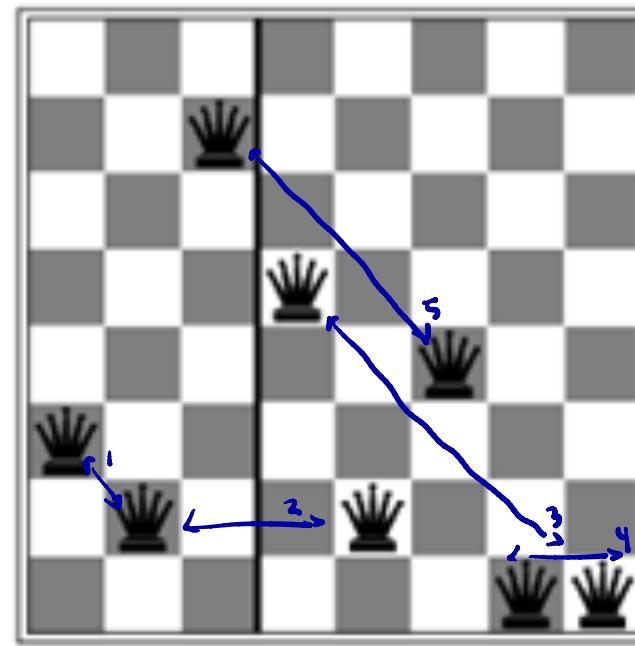
# State Value: Fitness Function

- Evaluation function (**fitness function**).
- evaluasi semua individu.*

- Higher values for better states.
- Fitness function for n-queen: number of non-attacking pairs of queens

$$\begin{aligned} \cdot \min &= 0 && \text{8 ratu bisa tidak serang 7.} \\ \cdot \max &= (8 \times 7)/2 = 28 && \text{1 wrong } 2 \text{ wrong } 1. \\ &&& \text{(global maximum)} \end{aligned}$$

32752411



$$F(32752411) = 28 - 5 = 23$$



# Initial Population & Successor Function

- GA starts with  $k$  randomly generated states.
- A successor state is generated by combining two parent states
- Produce the next generation of states by selection, crossover, and mutation



pendekatan roulette wheel  
threshold

# Random Selection of Parent States

Probability of random selection

24748552

- $24/(24+23+20+11)$
- = 31%

32752411

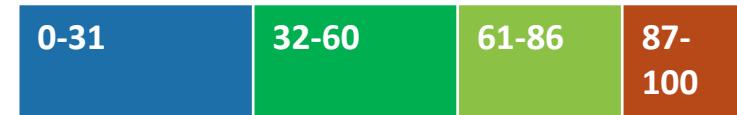
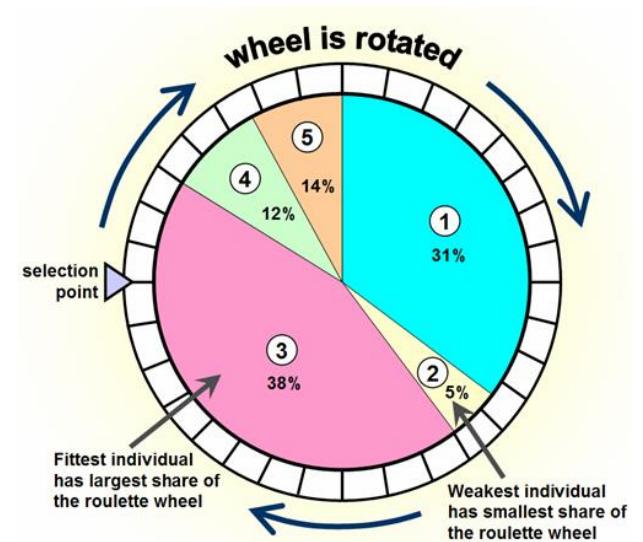
- $23/(24+23+20+11)$
- = 29%

24415124

- $20/(24+23+20+11)$
- = 26%

32543213

- $11/(24+23+20+11)$
- = 14%



random(0,100)

32752411

24748552

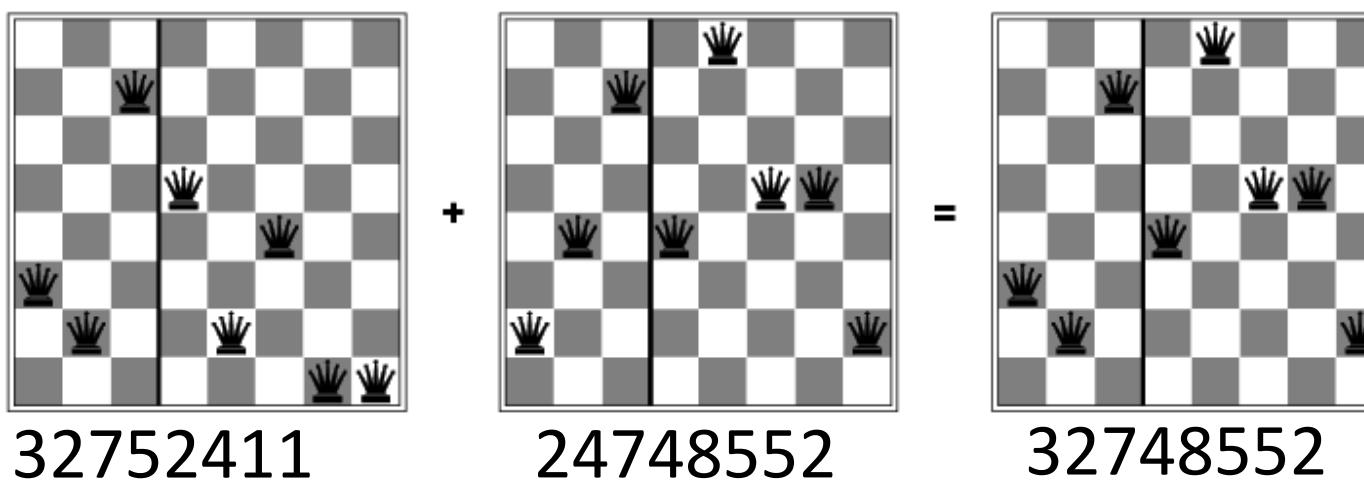
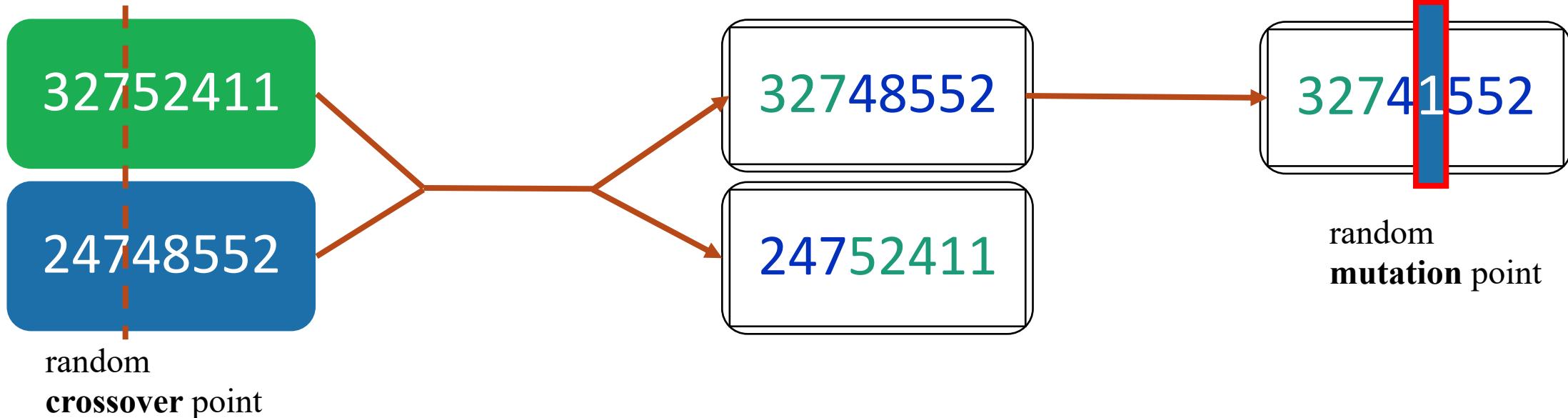
32752411

24415124



*bungantung ke probability juga*

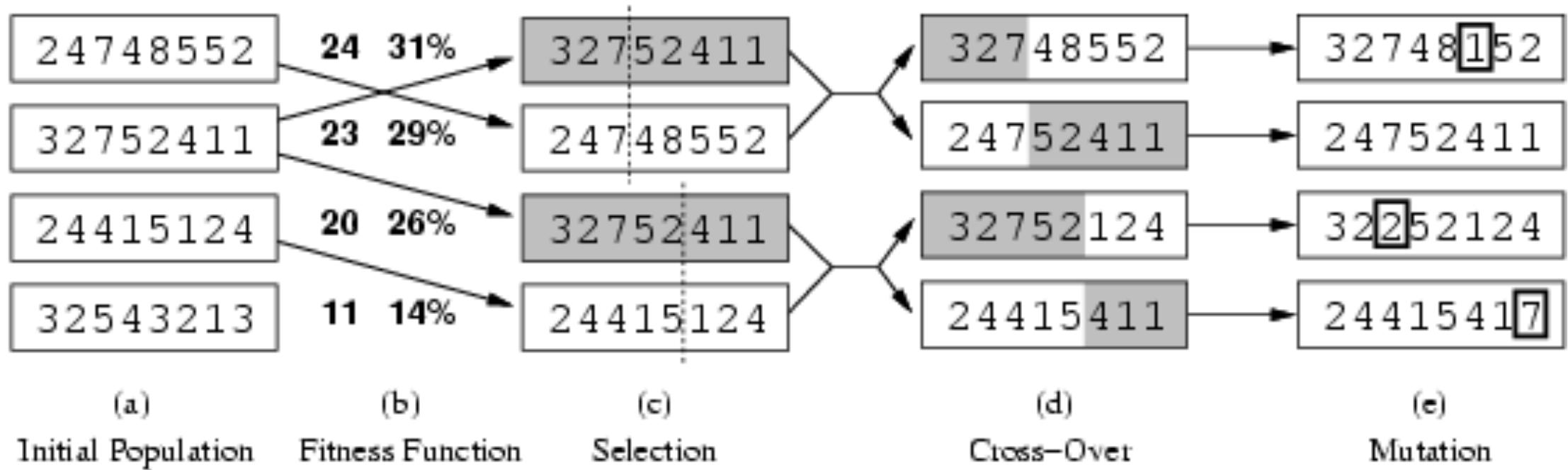
# Cross Over / Reproduce & Mutation



In more popular version, each mating of two parents produces only one offspring, not two (Russel & Norvig, 2010)



# Genetic Algorithm: Illustration (Russel & Norvig, 2010)



# Genetic Algorithm

**function** GENETIC-ALGORITHM(*population*, FITNESS-FN) **returns** an individual

**inputs:** *population*, a set of individuals

FITNESS-FN, a function that measures the fitness of an individual

**repeat**

*new\_population*  $\leftarrow$  empty set

**for** *i* = 1 **to** SIZE(*population*) **do**

*x*  $\leftarrow$  RANDOM-SELECTION(*population*, FITNESS-FN)

*y*  $\leftarrow$  RANDOM-SELECTION(*population*, FITNESS-FN)

*child*  $\leftarrow$  REPRODUCE(*x*, *y*)

**if** (small random probability) **then** *child*  $\leftarrow$  MUTATE(*child*)

add *child* to *new\_population*

*population*  $\leftarrow$  *new\_population*

**until** some individual is fit enough, or enough time has elapsed

**return** the best individual in *population*, according to FITNESS-FN



# Summary

$k$  randomly generated states (population:  $k$  individual)

Fitness function (state value)

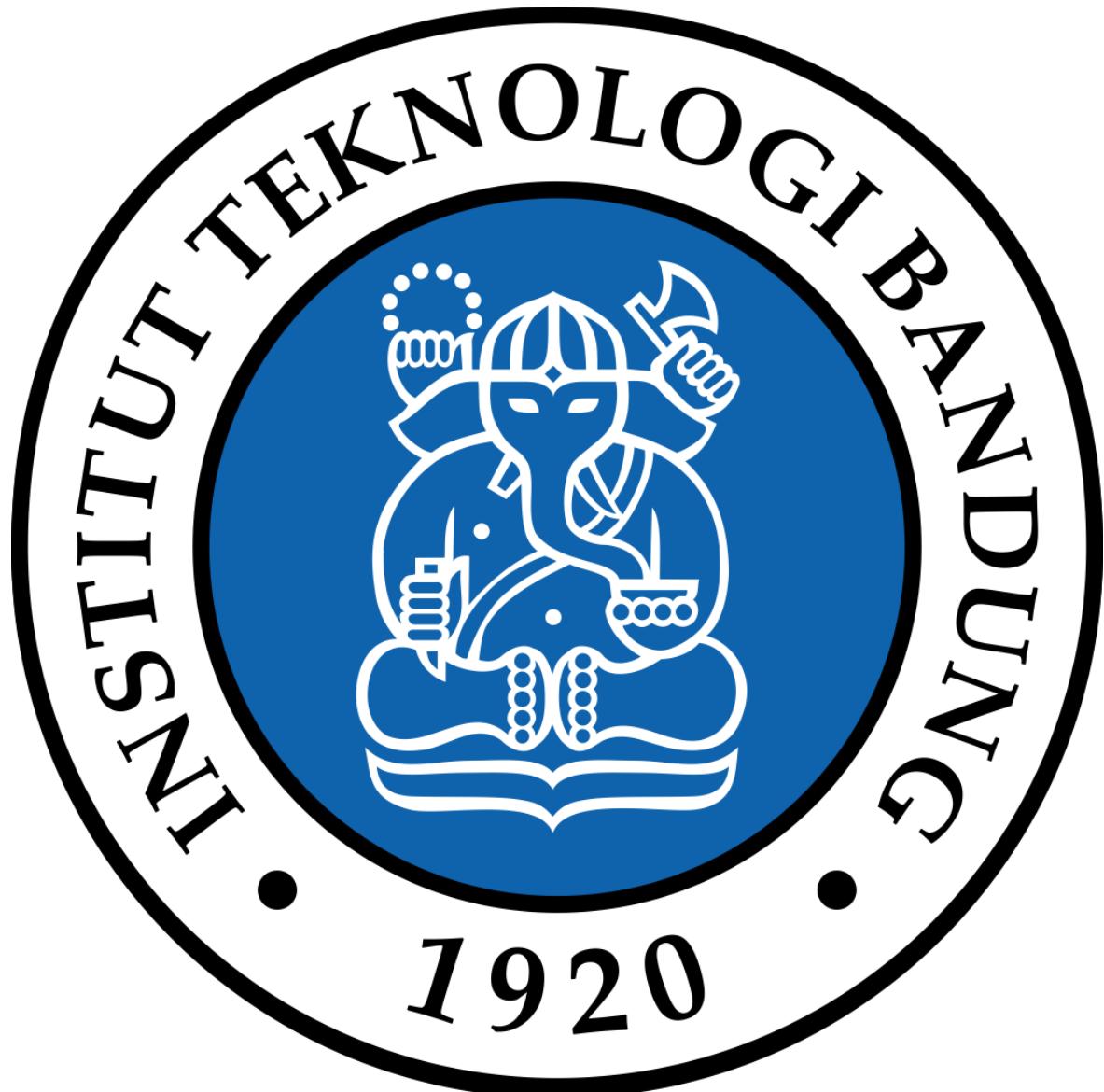
successor function:  
combining two parent states (selection, cross-over, mutation)

pilih teknik terbaik





# Terima Kasih



EDUNEX ITB



1. Teknik local search yang mengenumerasi semua state tetangga (successor) dan current state pindah ke state tetangga terbaik jika nilai evaluasi state tetangga terbaik lebih baik dari nilai evaluasi current state adalah:
- a. generate semua ~~tp ambil yg paling baik~~ → dia hanya mengevaluasi i state tetangga r ga semua.
  - b. hill-climbing with sideways move; c. stochastic hill-climbing
2. Simulated annealing (SA) merupakan kombinasi dari hill-climbing dan random walk. Manakah karakteristik dari SA ?
- a. jika penurunan temperatur T cukup lambat, SA menjamin akan menemukan optimum global dengan peluang mendekati 1. mengacil
  - b. SA memiliki peluang yang membesar secara eksponensial untuk memilih state tetangga yang lebih buruk nilai evaluasinya.
  - c. A membangkitkan secara random satu state tetangga, bukan semua state tetangga untuk setiap iterasinya.
3. Misalkan current state memiliki nilai evaluasi  $h$ , dan objektifnya mencari nilai  $h$  maksimum. Manakah perpindahan current state ke tetangga yang benar.
- a. Pada steepest-ascend hill climbing, current state akan selalu pindah ke tetangga terbaik dari semua tetangga yang ada, dengan nilai evaluasi  $h_{\text{next}}$  jika  $h_{\text{next}} > h$ . Jika  $h_{\text{next}} \leq h$ , masih ada peluang untuk berpindah state. → ga ada peluang pindah tetangga.
  - b. Pada simulated annealing, current state akan selalu pindah ke tetangga yang diambil secara acak dengan nilai evaluasi  $h_{\text{next}}$  jika  $h_{\text{next}} > h$ . Jika  $h_{\text{next}} \leq h$ , masih ada peluang untuk berpindah state.
  - c. Pada algoritma genetika, current state akan selalu pindah ke tetangga dengan nilai evaluasi  $h_{\text{next}}$  yang dihasilkan dari proses seleksi, cross-over, dan mutasi tanpa mempedulikan apakah  $h_{\text{next}} > h$ .

Pada simulated annealing, diberikan sejumlah kondisi nilai temperatur (T), current.value, dan neighbor.value.

Diketahui batas move probability adalah 0.6. Pada ketiga kasus berikut, tentukanlah peluang neighbor state akan

dipilih menggantikan current state, serta current state diset dengan apa.

(Nilai 10)

a. pindah ke neighbor

a. T = 10, current.value = 5, neighbor.value = 3 → tetangga is worse,  
e<sup>ΔE</sup> · e <sup>$\frac{3-5}{10}$</sup>  · e <sup>$\frac{-2}{10}$</sup>  · 0.8 am.

b. pindah ke neighbor

b. T = 3, current.value = 3, neighbor.value = 5 → tetangga is better.  
lgs o pindah.

c. T = 0, current.value = 5, neighbor.value = 3

→ ga ngapa²in

soalnya

T nya = 0, lgs o return parent.

- Pada algoritma genetika dengan populasi 4 individu (A,B,C,D), misalkan nilai fitness setiap individu secara berturut-turut adalah 10,8,7,5. Jika menggunakan roulette wheel dengan urutan individu tetap, lakukanlah proses seleksi pembentukan populasi untuk cross-over dengan urutan bilangan acak 0.55; 0.25; 0.85; 0.35

car probability

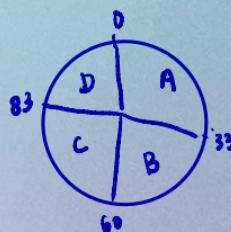
proportional fitness function

$$A: \frac{10}{30} = 0.33$$

$$B: \frac{8}{30} = 0.27$$

$$C: \frac{7}{30} = 0.23$$

$$D: \frac{5}{30} = 0.17$$



A	0-33	B	34-60	C	61-83	D	84-100
---	------	---	-------	---	-------	---	--------

$$0.55 \rightarrow B$$

$$0.25 \rightarrow A$$

$$0.85 \rightarrow D$$

$$0.35 \rightarrow B$$

