Week	Content	CO	Activity and Assessments			
			Quiz	HW	Presentation	UTS
20 Aug	1. Cakupan AI 2. What is CI 3. Pengenalan Algoritma Genetika	CO1				5
27 Aug	1. Teorema skema holland 2. Pengkodean Algoritma Genetika 3. Formulasi Fungsi Fitness	CO2	2,5			5
03 Sep	1. Project (proposal) 2. Proses Seleksi	CO2	2,5		5	
10 Sep	Proses Crossover	CO2	5			5
17 Sep	1. Proses Mutasi 2. Update Generasi 3. Parameter Algoritma Genetika	CO2				5
24 Sep	ACO dan PSO	CO2		5		
01 Oct	Project (Presentation)	CO2		5	5	
		Total	10	10	10	20

Week	Content	CO	Activity and Assessments			
			Quiz	HW	Presentation	UAS
22 Oct	What is Machine Learning	CO3				5
29 Oct	Linear classification dan regression	CO4	2,5			2,5
05 Nov	Traditional Machine Learning	CO4	2,5			2,5
12 Nov	Project	CO5			5	
19 Nov	Deep Learning	CO5	2,5			5
26 Nov	Metode dan Matrik Evaluasi	CO6	2,5			5
03 Dec	Project	CO5		10	5	
		Total	10	10	10	20

UNIVERSITAS
GADJAH MADA

Genetic Algorithm: Crossover

Aina Musdholifah, S.Kom., M.Kom, Ph.D.

How to solve problem using GA ?

UNIVERSITAS GADJAH MADA

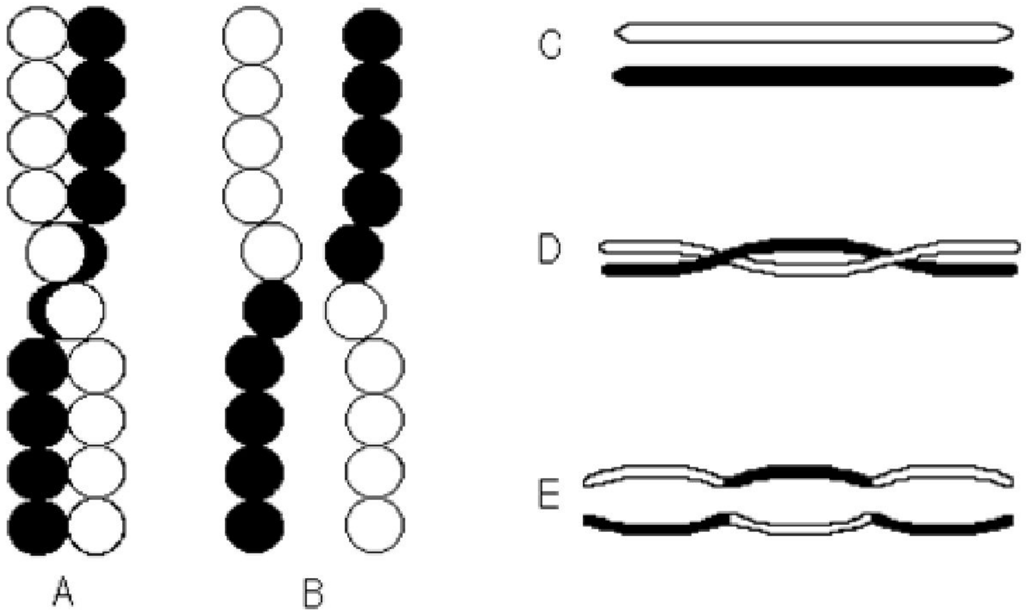
- Generate Population ... done
- Encoding $\longrightarrow$ Chromosome or individu … done
- Define Fitness Function … done
- Define GA Operator
    - Selection … done
    - **Cross Over**
    - Mutation

Crossover

- Taking partial / alela sequence fragments of DNA genome to be incorporated into the offspring

- Crossover operator in GA represent what happens in the real world

- Many methods have been proposed
- Each method has specific characteristics and may be only be used on certain types of representation. For example, cross over to the permutation representation is specific to the problem in permutation.



Crossover for

Binary Representation

Integer Representation

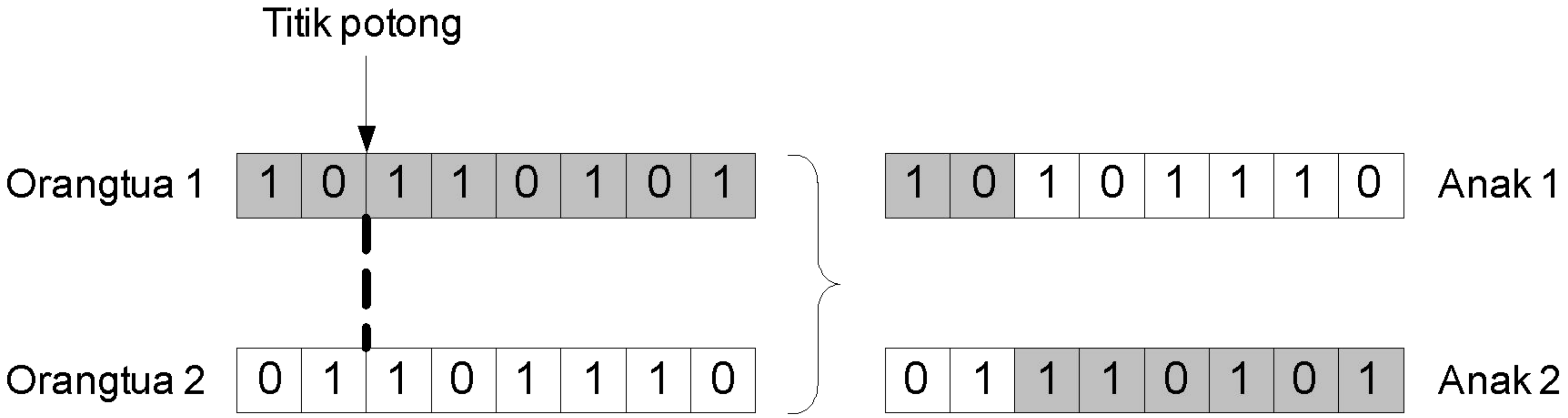
Real Representation

Permutation Representation

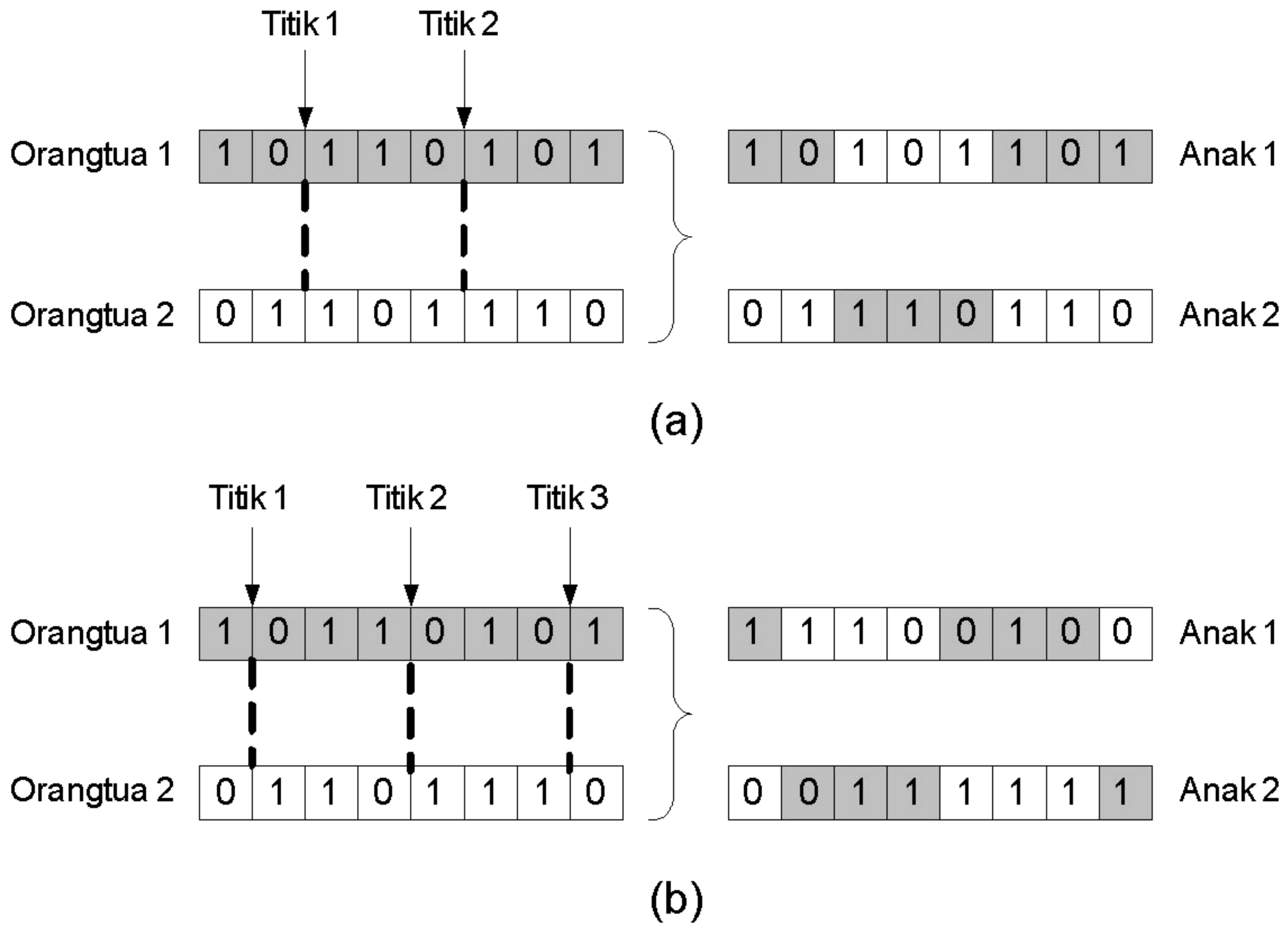
Crossover for Binary Representation

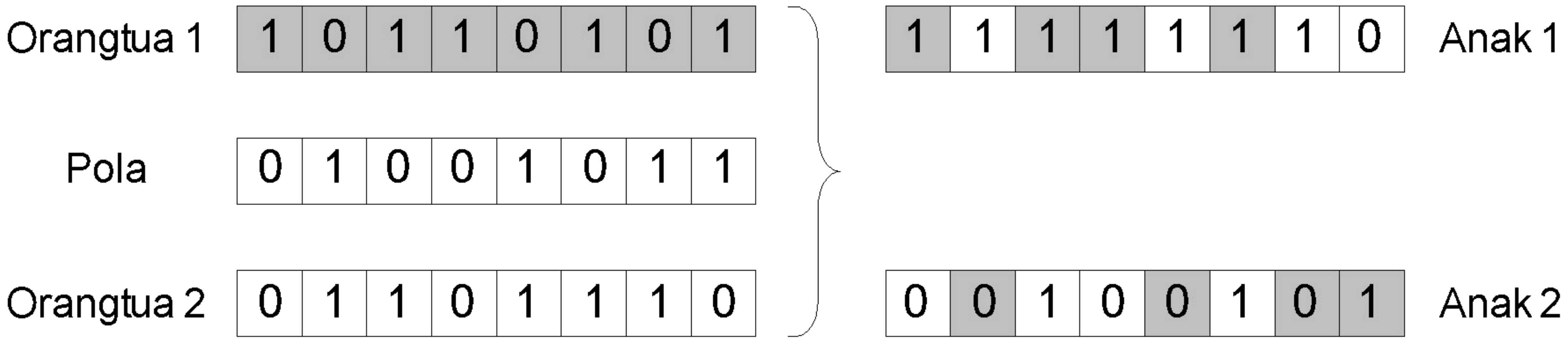
- 1-point crossover
- Multipoint crossover
- Uniform crossover

1-point crossover

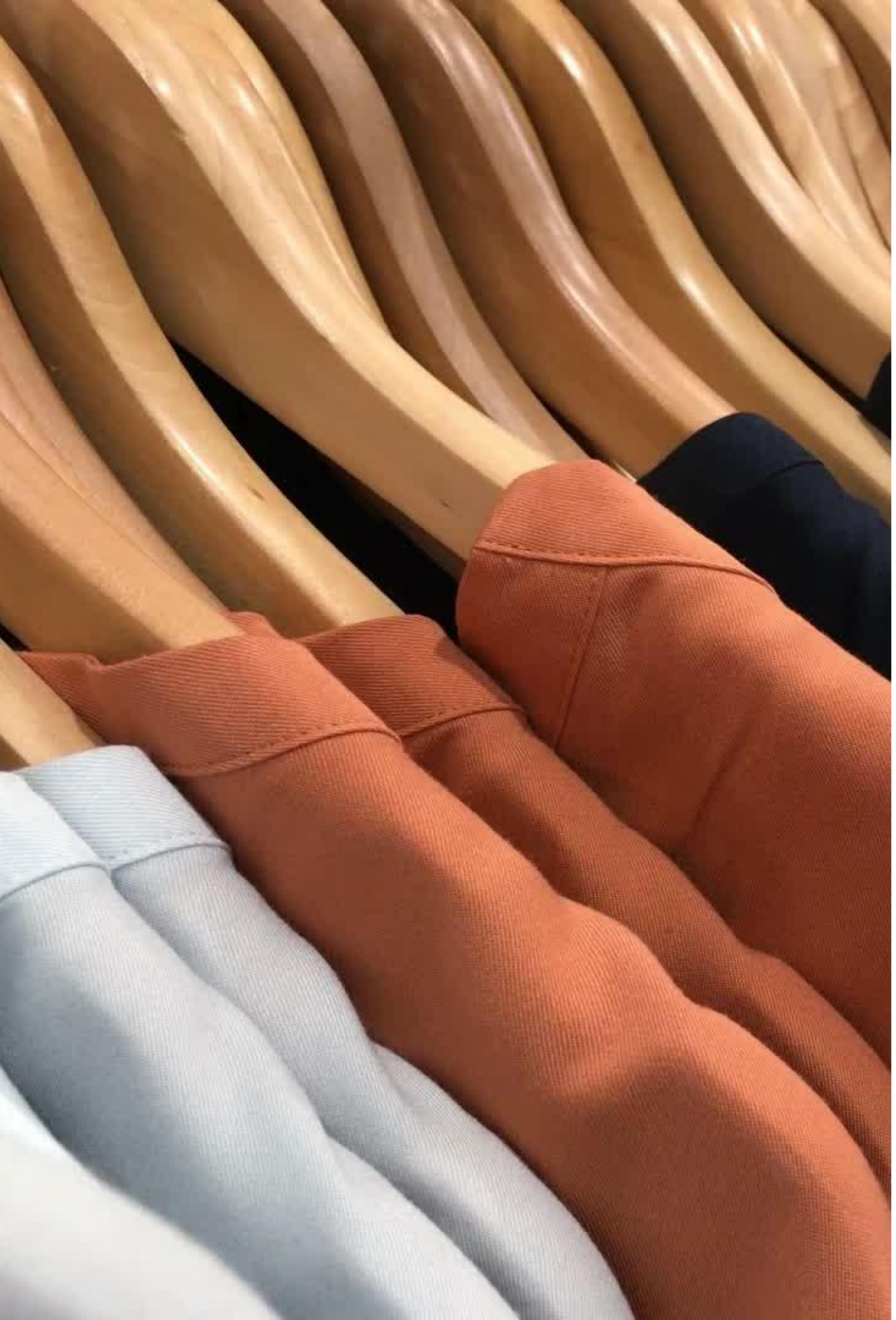


Multipoint crossover





Uniform Crossover



uniform crossover

P1 : 0 0 0 0 0 0 0 0
P2 : 1 1 1 1 1 1 1 1

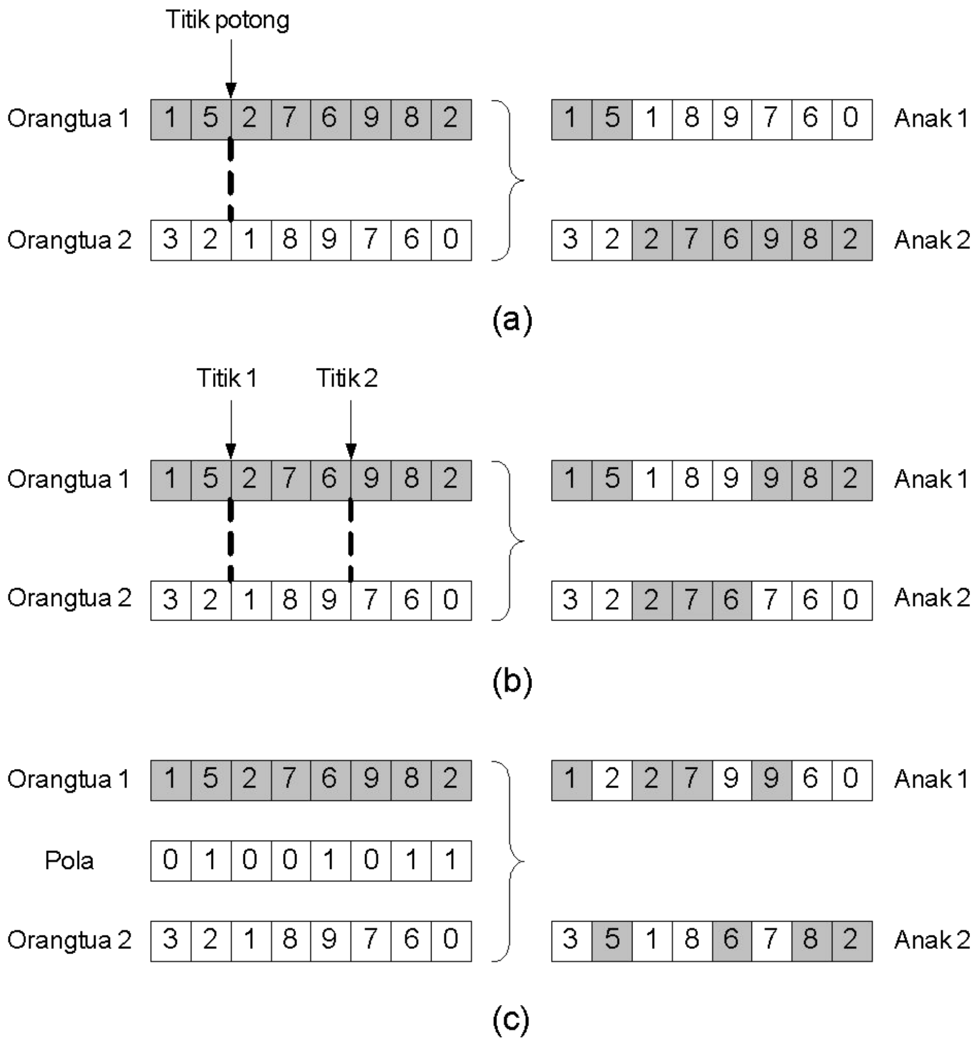
Uniform Crossover with pattern 00100110

Off 1: 0 0 1 0 0 1 1 0
Off 2: 1 1 0 1 1 0 0 1

Crossover For Integer Representation

- 1-point crossover
- Multipoint crossover
- uniform crossover

Crossover For Integer Representation



Crossover For Real Representation

UNIVERSITAS GADJAH MADA

We can understand that the three methods Cross over in previous slide can be used for binary and integer representation.

This is due to both representations have the values of the genes in a finite interval. The values of the genes in the binary representation is 0 or 1.

On the representation of integers, may be there is little variation in the value for each gene. For example, integer representation used to represent real numbers only have 10 possible values for each genes.

What if each gene may have a value that has real representation? For a real representation, cross over can be done in two ways: **discrete** and **intermediate**

Crossover For Real Representation: **Discrete**

The methods to select position of genes is Cross over multi points or Cross over uniform.

Each gene in offspring z derived from one parent (x, y) with equal probability, $z_i = x_i$ **or** y_i.

Crossover For Real Representation: **Intermediate**

- Using the idea of child development in the form of chromosomes "between" from both parents.
- Therefore, Cross over in this type called as arithmetic crossover.
- Each gene in children z^1 and z^2 are obtained by the formula :

  $z_i^1 = \alpha p_i^2 + (1-\alpha)p_i^1$ and $z_i^2 = \alpha p_i^1 + (1-\alpha)p_i^2$

  in which $0 \leq \alpha \leq 1$.

Parameter α can be constant (*uniform arithmetical crossover*), variable (for instance, depend on population age), or can be define randomly in any time.

Cross over *Intermediate*

- *single arithmetic crossover,*
- *simple arithmetic crossover*, and
- *whole arithmetic crossover*.

Single arithmetic crossover

Suppose two parent of chromosomes expressed as:

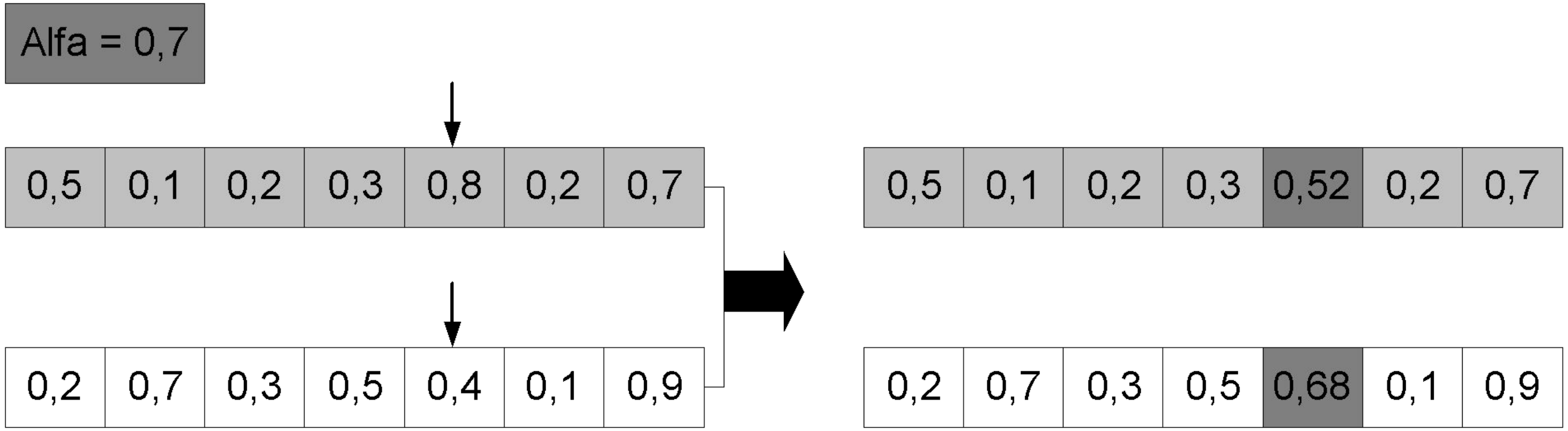
$$\langle x_1,\ldots,x_n \rangle \qquad \langle y_1,\ldots,y_n \rangle$$

Select 1 gen randomly, for example *k*. Then, both the child will be produced by:

$$\text{Anak 1}: \langle x_1,\ldots,x_{k-1}, \alpha y_k + (1-\alpha)x_k,\ldots,x_n \rangle$$

$$\text{Anak 2}: \langle y_1,\ldots,y_{k-1}, \alpha x_k + (1-\alpha)y_k,\ldots,y_n \rangle$$

Single arithmetic crossover



$$\text{Anak}\,1:\langle x_1,\ldots,x_{k-1},\alpha y_k+(1-\alpha)x_k,\ldots,x_n\rangle$$

$$\text{Anak}\,2:\langle y_1,\ldots,y_{k-1},\alpha x_k+(1-\alpha)y_k,\ldots,y_n\rangle$$

Simple arithmetic crossover

Suppose two parent of chromosomes expressed as:

$$\langle x_1,\ldots,x_n \rangle \qquad \langle y_1,\ldots,y_n \rangle$$

Select 1 gen randomly, for example *k*. Then, both the child will be produced by:

$$\text{Anak}\,1 : \langle x_1,\ldots,x_k, \alpha y_{k+1} + (1-\alpha)x_{k+1},\ldots,\alpha y_n + (1-\alpha)x_n \rangle$$

$$\text{Anak}\,2 : \langle y_1,\ldots,y_k, \alpha x_{k+1} + (1-\alpha)y_{k+1},\ldots,\alpha x_n + (1-\alpha)y_n \rangle$$

Simple arithmetic crossover

Alfa = 0,5

0,5	0,1	0,2	0,3	0,8	0,2	0,7
0,2	0,7	0,3	0,5	0,4	0,1	0,9

0,5	0,1	0,2	0,3	0,8	0,15	0,8
0,2	0,7	0,3	0,5	0,4	0,15	0,8

$$\text{Anak 1}: \langle x_1, \ldots, x_k, \alpha y_{k+1} + (1-\alpha) x_{k+1}, \ldots, \alpha y_n + (1-\alpha) x_n \rangle$$

$$\text{Anak 2}: \langle y_1, \ldots, y_k, \alpha x_{k+1} + (1-\alpha) y_{k+1}, \ldots, \alpha x_n + (1-\alpha) y_n \rangle$$

Whole arithmetic crossover

Suppose two parent of chromosomes expressed as:

$\langle x_1,\ldots,x_n \rangle$ $\langle y_1,\ldots,y_n \rangle$

$z_i^1 = \alpha y_i + (1-\alpha)x_i$ and $z_i^2 = \alpha x_i + (1-\alpha)y_i$

Whole arithmetic crossover

$$z_i^1 = \alpha p_i^2 + (1-\alpha)p_i^1$$

and

$$z_i^2 = \alpha p_i^1 + (1-\alpha)p_i^2$$

Alfa = 0,5

0,5	0,1	0,2	0,3	0,8	0,2	0,7
0,2	0,7	0,3	0,5	0,4	0,1	0,9

0,35	0,4	0,25	0,4	0,6	0,15	0,8
0,35	0,4	0,25	0,4	0,6	0,15	0,8

Crossover For Permutation Representation

- Cross over to permutation representation will be used to solve permutation problem, for example *Travelling Salesman Problem* (TSP)

- In TSP, permutation representation using gen position as the order of the city. Cross over to binary representation, integer representation and real representation can not be used for this representation.

- Why not?

Crossover for Permutation Representation

- *Order crossover*
- *Partially mapped crossover*
- *Cycle crossover*
- *Edge Recombination*

Permutation Representation: **Order crossover**

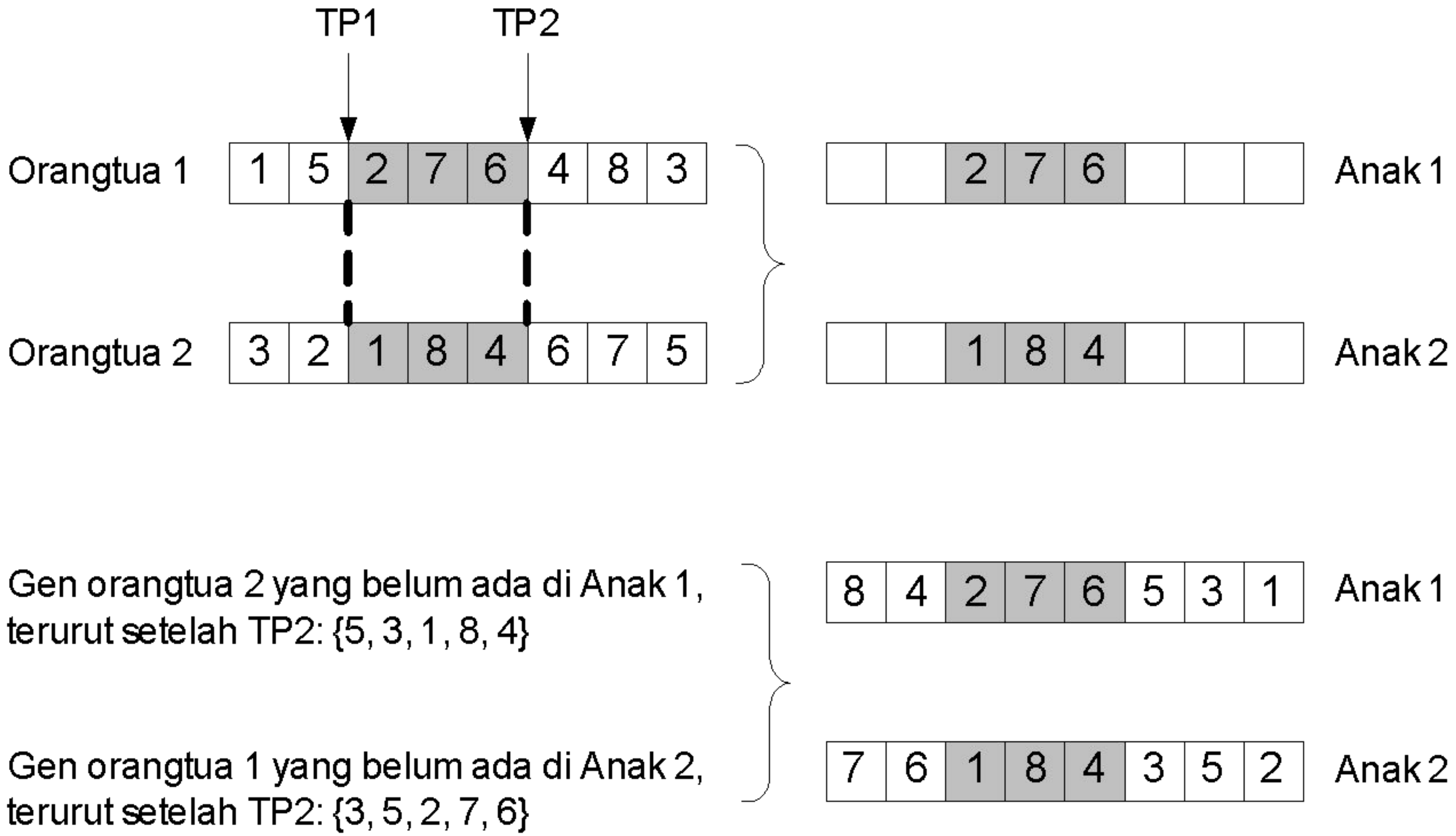
UNIVERSITAS GADJAH MADA

Observes that order is important, not necessarily position.

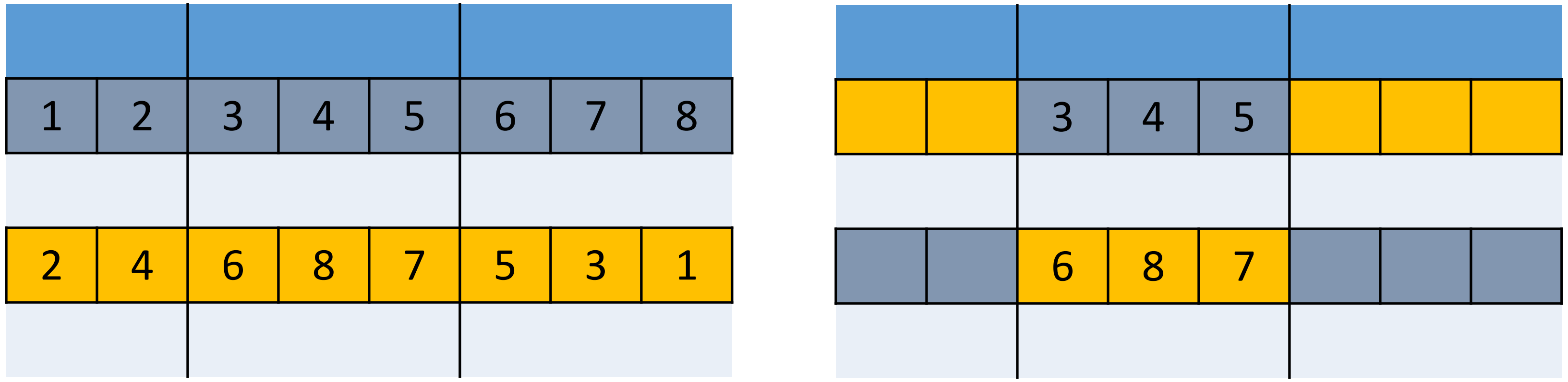
How it works:

- Choose 2 cut points
- Copy between cut points to offspring
- Starting from 2nd cut point in one parent, fill missing cities in order they appear in another parent

Order crossover (illustration)



example



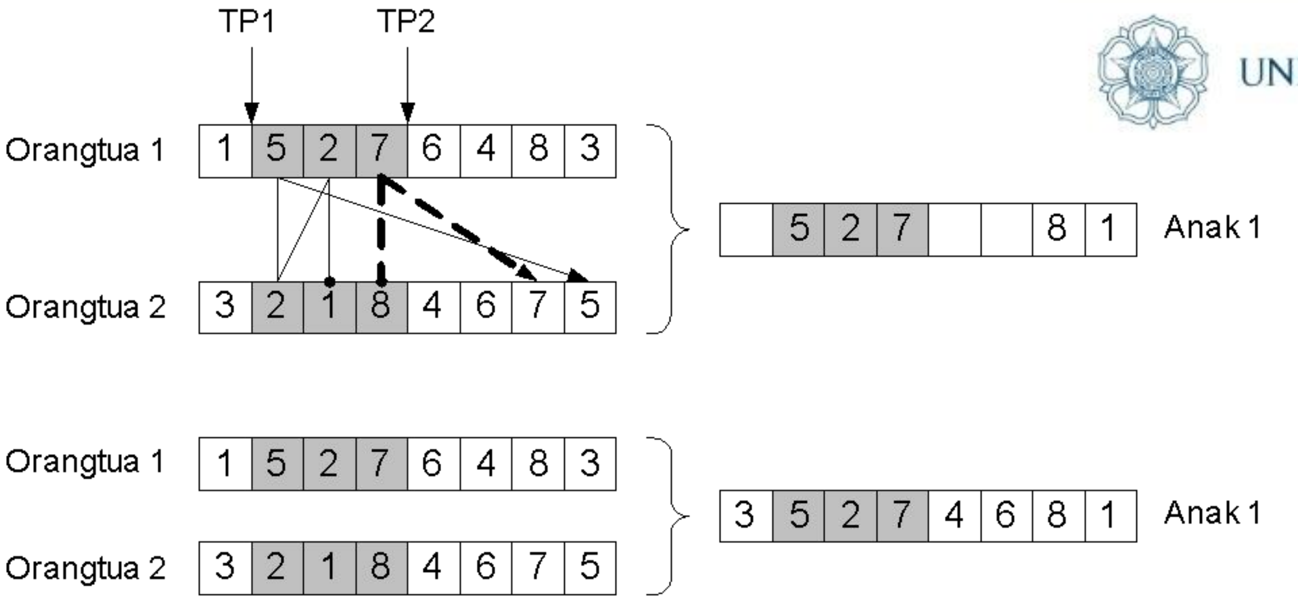
Partially mapped crossover

Portion of one parent is mapped to a portion of another parent

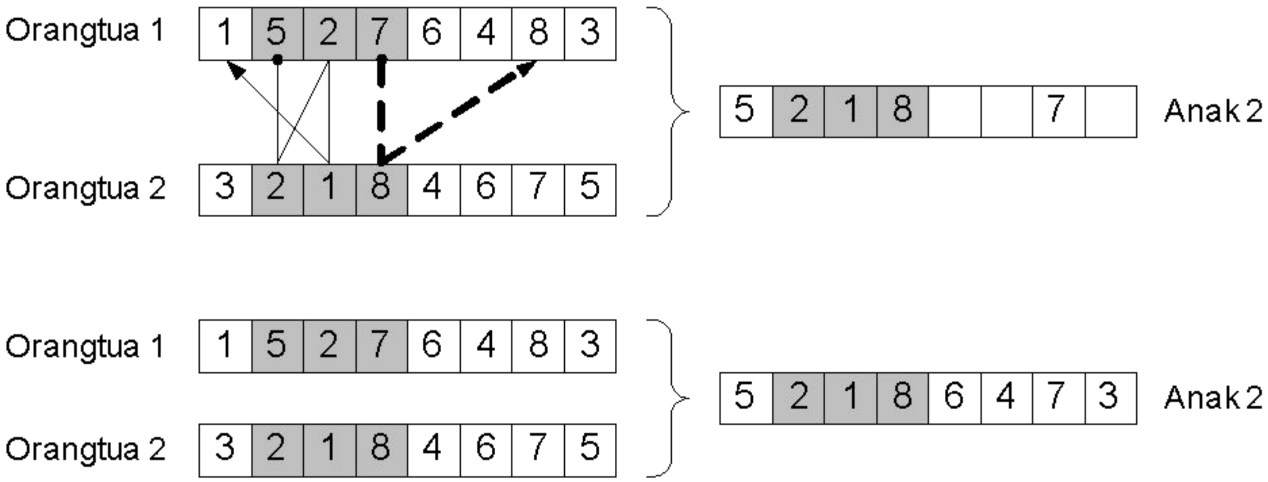
Remaining info is exchanged

How it works:

1. Choose two random cut points
2. Define mapping
3. Copy mapping section (between cut points) to offspring
4. Fill in remainder of offspring using mapping


TP1
TP2
Orangtua 1
1 5 2 7 6 4 8 3
Orangtua 2
3 2 1 8 4 6 7 5
5 2 7 8 1
Anak 1
Orangtua 1
1 5 2 7 6 4 8 3
Orangtua 2
3 2 1 8 4 6 7 5
3 5 2 7 4 6 8 1
Anak 1

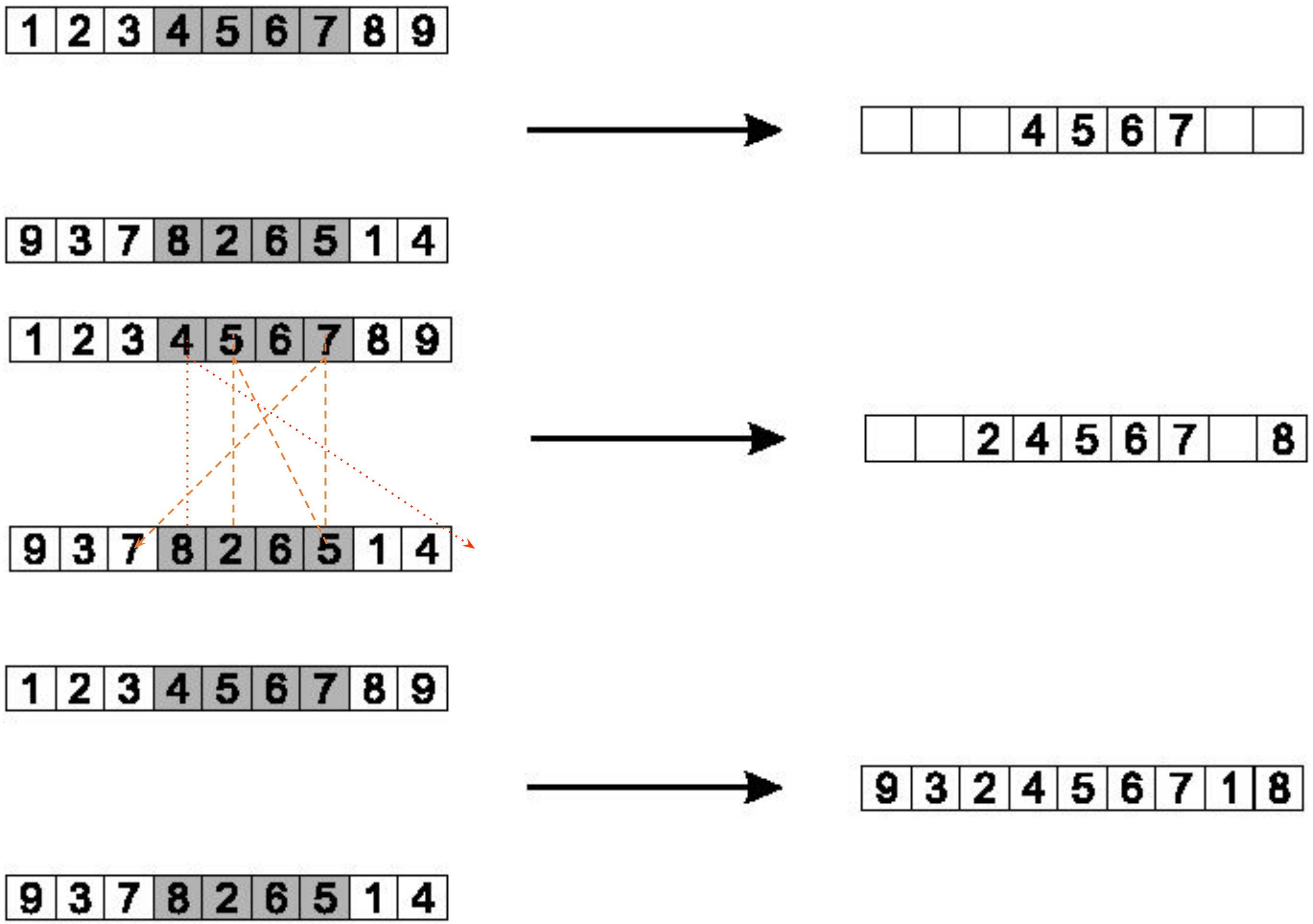
(a)


Orangtua 1
1 5 2 7 6 4 8 3
Orangtua 2
3 2 1 8 4 6 7 5
5 2 1 8 7
Anak 2
Orangtua 1
1 5 2 7 6 4 8 3
Orangtua 2
3 2 1 8 4 6 7 5
5 2 1 8 6 4 7 3
Anak 2

(b)

PMX example

- Step 1
- Step 2
- Step 3



Another example of pmx co

Cut Point 1 *Cut Point 2*

P1 = [2 1 | 3 4 5 | 6 7]
P2 = [4 3 | 1 2 5 | 7 6]

O1 = [4 3 1 2 5 6 7]
O2 = [2 1 3 4 5 7 6]

P1 = (1 2 3 | 4 5 6 7 | 8 9)

P2 = (4 5 2 | 1 8 7 6 | 9 3)

O1 = (4 2 3 | 1 8 7 6 | 5 9)

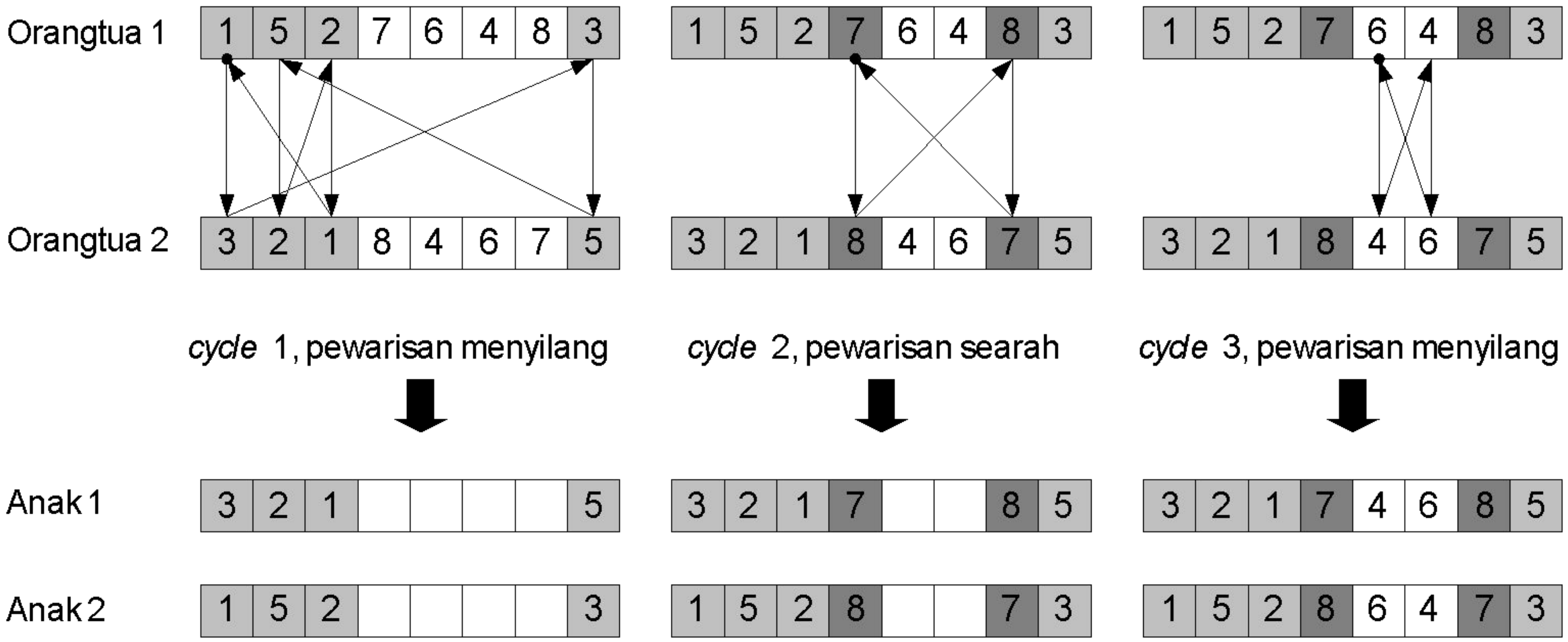
O2 = (1 8 2 | 4 5 6 7 | 9 3)

Cycle crossover

- Creates an offspring where every position is occupied by a corresponding element from one of the parents using two cycles, crossed and the direction alternately.

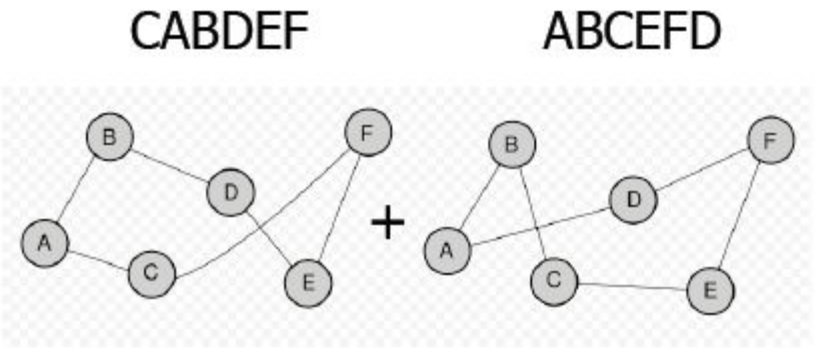
- The Cycle Crossover operator identifies a number of so-called cycles between two parent chromosomes. Then, to form Child 1, cycle one is copied from parent 1, cycle 2 from parent 2, cycle 3 from parent 1, and so on.

Cycle crossover



Edge Crossover

- Creates a path (offspring) that is similar to a set of existing paths (parents) by looking at the edges rather than the vertices.
- Edge Map
  - For each node, gives list of other nodes to which it has an edge in either parent



A: B C D
B: A C D
C: A B E F
D: A B E F
E: C D F
F: C D E

Algorithm of *Edge* Crossover

Algorithm

- Let K be the empty list
- Let N be the first node of a random parent.

- While Length(K) < Length(Parent):
  - K := K + N (append N to K)
  - Remove N from all neighbor lists

  - If N's neighbor list is non-empty then
    - let N* be the neighbor of N with the fewest neighbors in its list (or a random one, should there be multiple)
  - else let N* be a randomly chosen node that is not in K
  - N := N*

Orangtua 1

1	5	2	7	6	4	8	3

Orangtua 2

3	2	1	8	4	6	7	5

UNIVERSITAS GADJAH MADA

Pencarian *edge* untuk semua gen pada kedua Orangtua menghasilkan tabel sbb:

Elemen	*Edge*
1	3, 5, 2, 8
2	5, 7, 3, 1
3	8, 1, 5, 2
4	6+, 8+
5	1, 2, 7, 3
6	4+, 7+
7	2, 6+, 5
8	4+, 3, 1

Tabel harus di- *update* setelah suatu elemen terpilih. Setelah 3 terpilih, semua *edge* 3 dihapus dari tabel.

Elemen	*Edge*
1	5, 2, 8
2	5, 7, 1
3	8, 1, 5, 2
4	6+, 8+
5	1, 2, 7
6	4+, 7+
7	2, 6+, 5
8	4+, 1

...

Pilihan	Elemen terpilih	Alasan	Hasil
Semua	3	Pemilihan secara acak	{3}
8, 1, 5, 2	8	Daftar *edge* terpendek	{3,8}
4, 1	4	Daftar *edge terpendek*	{3,8,4}
6	6	*Common edge*	{3,8,4,6}
7	7	*Common edge*	{3,8,4,6,7}
2, 5	2	Pemilihan acak	{3,8,4,6,7,2}
5, 1	1	Pemilihan acak	{3,8,4,6,7,2,1}
5	5	Hanya ada satu pilihan	{3,8,4,6,7,2,1,5}

Offspring-1.

The offspring-2 is produced in the same way. Because it is random, The offspring-2 can be different from The offspring-1.

Next…

- Mutation