

# GENETIC ALGORITHM



---

Bui Hong Ngoc



# Mục Lục

---

1. Tổng Quan - Evolutionary Algorithm
2. Genetic Algorithm - Giải Thuật
  - 2.1. Sơ đồ thuật toán
  - 2.2. Mã hóa
  - 2.3. Toán tử chọn lọc
  - 2.4. Toán tử lai ghép
  - 2.5. Toán tử đột biến
3. Genetic Programming - Sự khác biệt
4. Xây dựng



# Evolutionary Algorithm

---

- Có thể coi như một thuật toán tìm kiếm giải quyết các bài toán tối ưu.
- Metaheuristic: Một phương pháp heuristic để chọn các phương pháp heuristic
- Lấy cảm hứng từ tự nhiên, dựa trên nguyên lý tiến hóa



# Evolutionary Algorithm

---

Gồm một số nhóm chính

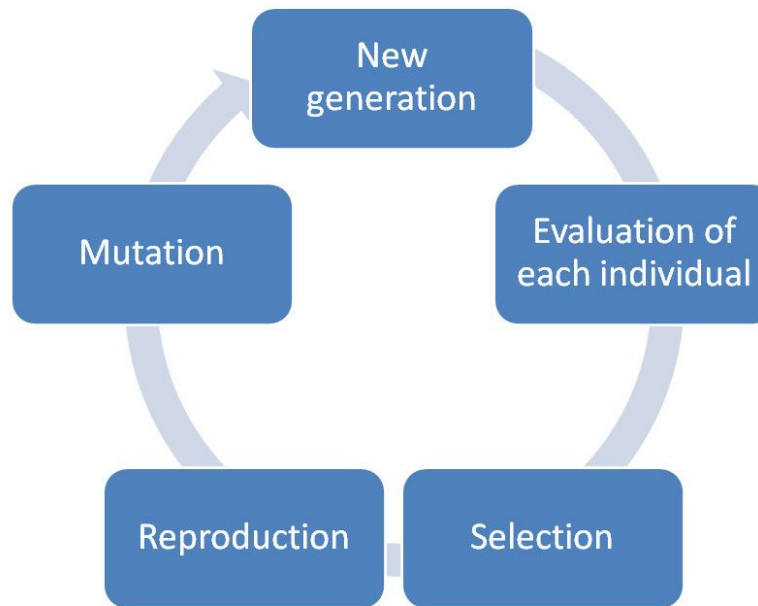
- Genetic Algorithm
- Genetic Programming
- Evolutionary Programming
- Gene Expression Programming



# Genetic Algorithm

---

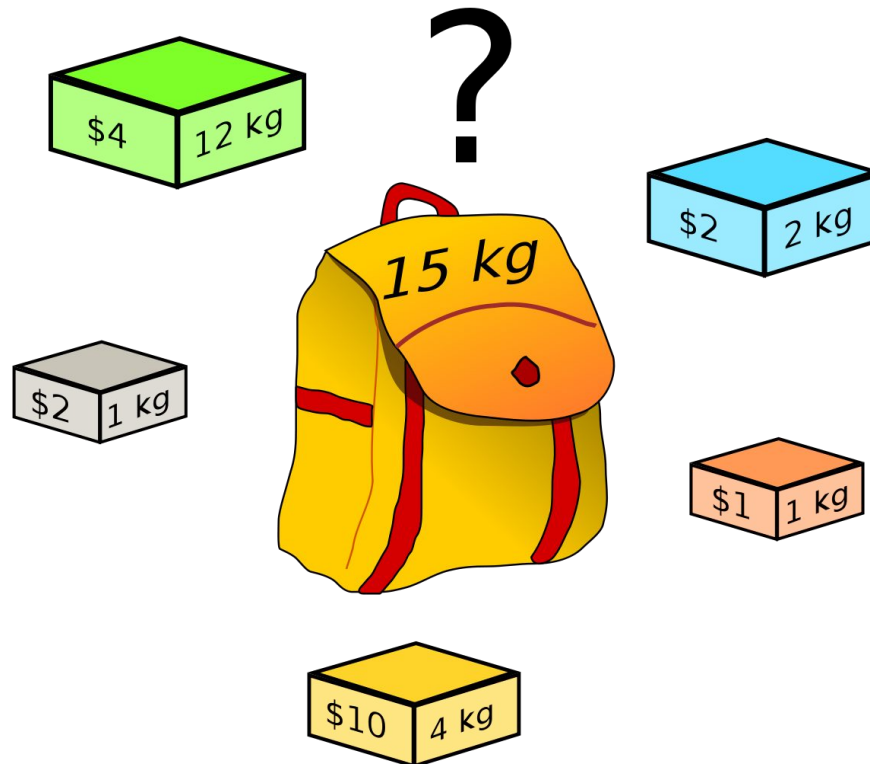
- Là một phân nhóm trong EA
- Dựa trên ý tưởng chọn lọc tự nhiên



# Genetic Algorithm

Được sử dụng để giải quyết các bài toán tối ưu

- VD: bài toán Knapsack



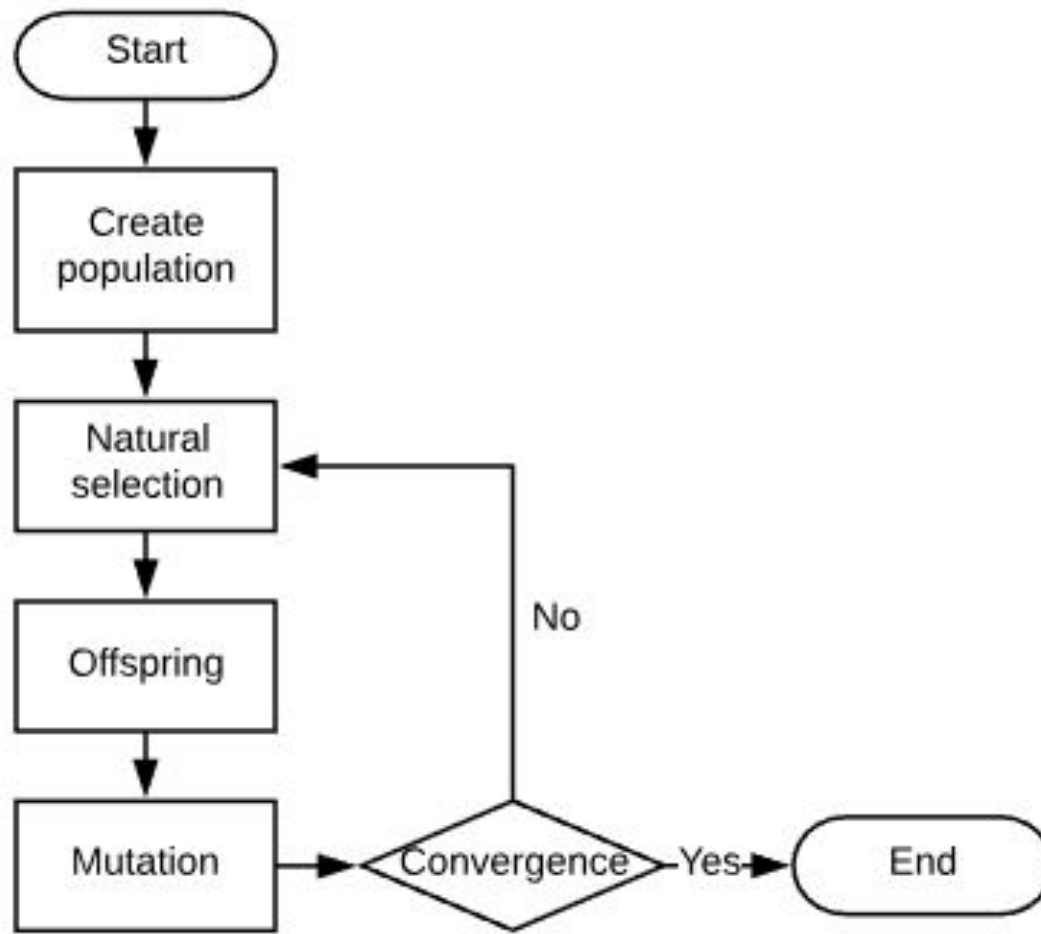


# Khái niệm cơ bản

---

- **Cá thể:** Là một đại diện cho lời giải của bài toán.
  - Trong ví dụ Knapsack là: Một cách chọn các vật vào balo
- **Quần thể:** Là một tập hợp các lời giải. Quần thể thuật toán tạo ra là không gian lời giải của bài toán mà thuật toán nhìn thấy
  - Trong ví dụ Knapsack: là không gian con của không gian lời giải cho bài toán Knapsack
- **Nhiệm Sắc Thể:** là biểu diễn mã hóa của cá thể. Liên kết giữa cá thể và nhiệm sắc thể là liên kết 1-1.

# Sơ đồ chi tiết thuật toán



Source: Daniel Gutierrez-Navarro





# Mã hóa - giải mã

---

- **Mã hóa:** Là quá trình chuyển từ cá thể sang nhiễm sắc thể của nó
- **Dạng mã hóa:**
  - Mã hóa nhị phân
  - Mã hóa hoán vị
  - Mã hóa Prufer



# Mã hóa nhị phân

---

- **Nhiệm sắc thể được biểu diễn dưới dạng 01**
  - VD: Với bài toán Knapsack:
  - Phân tử 1 ở vị trí  $i$  với ý nghĩa item  $i$  được chọn và ngược lại

Chromosome A	10110010110011100101
Chromosome B	11111110000000011111



## Mã hóa hoán vị

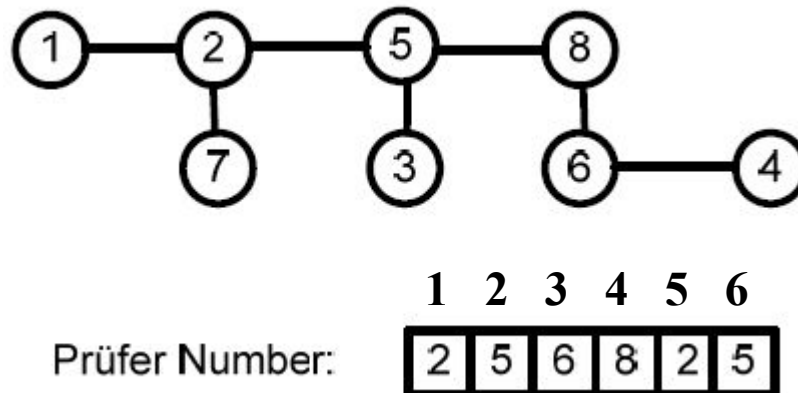
- **Nhiệm sắc thể được biểu diễn dưới dạng hoán vị của 1 tập hợp, thường là từ 1->n**
  - VD: Với bài toán TSP:
  - Một lời giải biểu diễn một chu trình của người di chuyển.

1 -> 5 -> 3 -> 2 -> 6 -> 7 -> 9 -> 8 -> 1

Chromosome A	1 5 3 2 6 4 7 9 8
Chromosome B	8 5 6 7 2 3 1 4 9

# Mã hóa Prufer

- Thường được sử dụng trong đồ thị, Nhiệm vụ thể biểu diễn cây trong đồ thị.
  - VD: Với cây sau:





# Toán tử chọn lọc

---

- Chọn ra một số lượng cá thể trong quần thể để đem đi sinh sản
- Chiến lược chọn rất quan trọng



# Toán tử chọn lọc

---

- **Random selection:** Chọn ngẫu nhiên  $n$  cá thể trong quần thể
- **Linear ranking selection:** Sắp xếp các cá thể theo thứ hàm thích nghi, cá thể tốt nhất có  $\text{rank1} = n/NP$ , cá thể tồi nhất có  $\text{rankn} = 1/NP$ .

Xác xuất để một cá thể được chọn:

$$p[i] = \text{rank}_i / \text{sum}(\text{rank1} + \dots + \text{rankn})$$

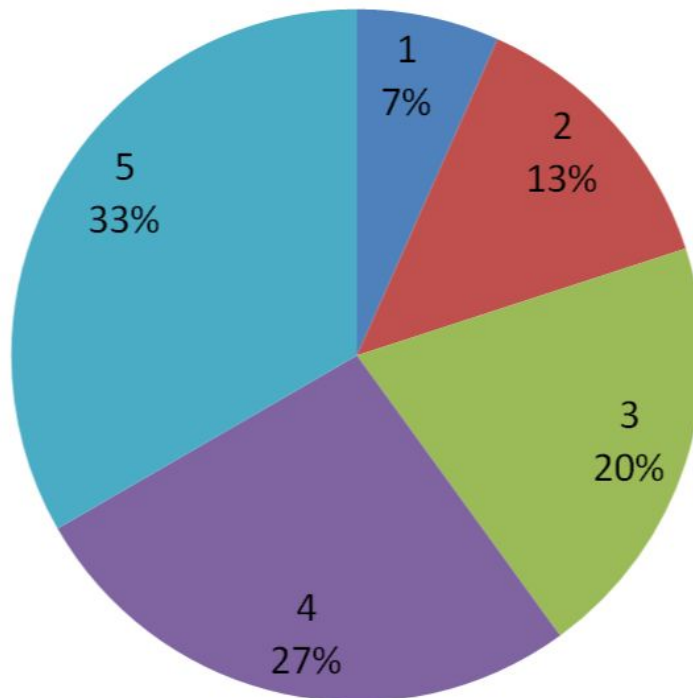
- **Exponential selection:** Tương tự như Linear ranking nhưng dùng hàm mũ cho rank.

$$p[i] = \text{rank}_i^c / \text{sum}(\text{rank1}^c + \dots + \text{rankn}^c)$$

# Toán tử chọn lọc

- **Roulette wheel selection:** sử dụng ngay hàm fitness của cá thể để tính xác suất được chọn của cá thể  
Xác suất để một cá thể được chọn:

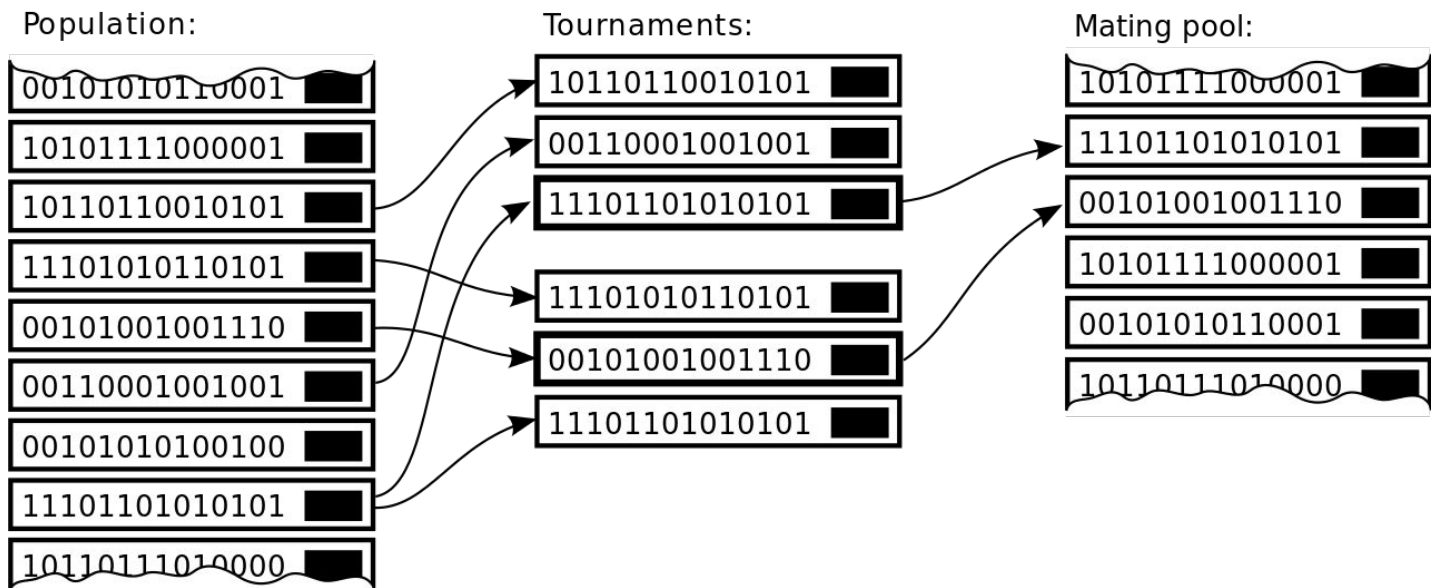
$$p[i] = \text{fitness}(i) / \text{sum}(\text{fitness}(1) + \dots + \text{fitness}(n))$$



Individual	Fitness
1	1.0
2	2.0
3	3.0
4	4.0
5	5.0

# Toán tử chọn lọc

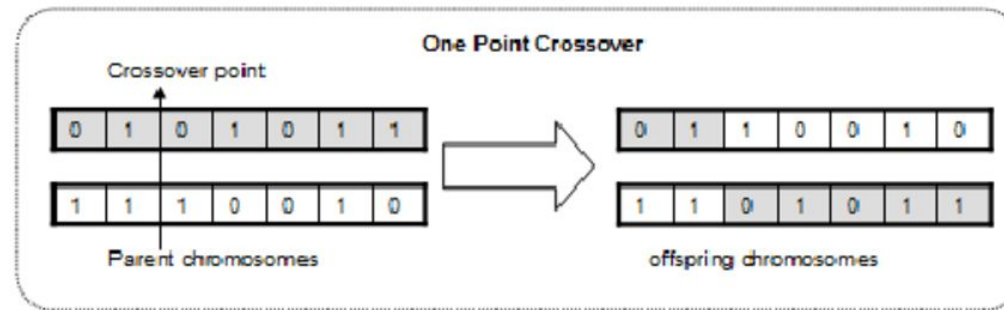
- **Tournament selection:** Chọn ngẫu nhiên X cá thể trong quần thể, cho chúng cạnh tranh với nhau, lấy cá thể mạnh nhất  
Lặp lại n lần để được n cá thể.



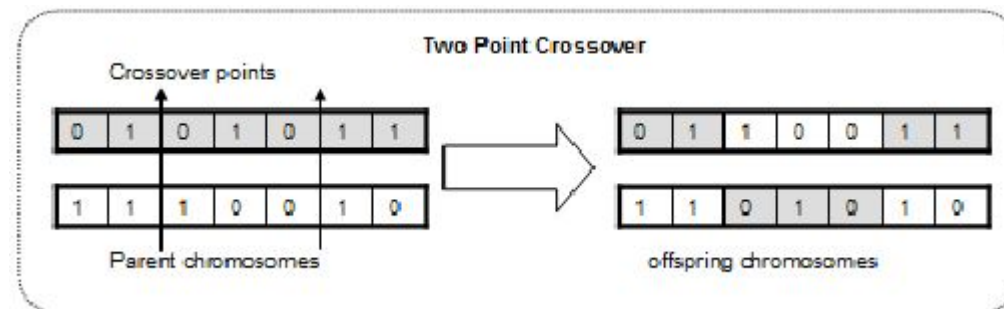


# Toán tử lai ghép - Mã hóa nhị phân

- Lai ghép một điểm cắt



- Lai ghép hai điểm cắt





# Toán tử lai ghép - Mã hóa hoán vị

---

- Order 1 crossover

Parent 1: 8 4 7 3 6 2 5 1 9 0

Parent 2: 0 ~~1~~ ~~2~~ ~~3~~ 4 ~~5~~ ~~6~~ 7 8 9

Child 1: 0 4 7 3 6 2 5 1 8 9

# Toán tử lai ghép - Mã hóa hoán vị

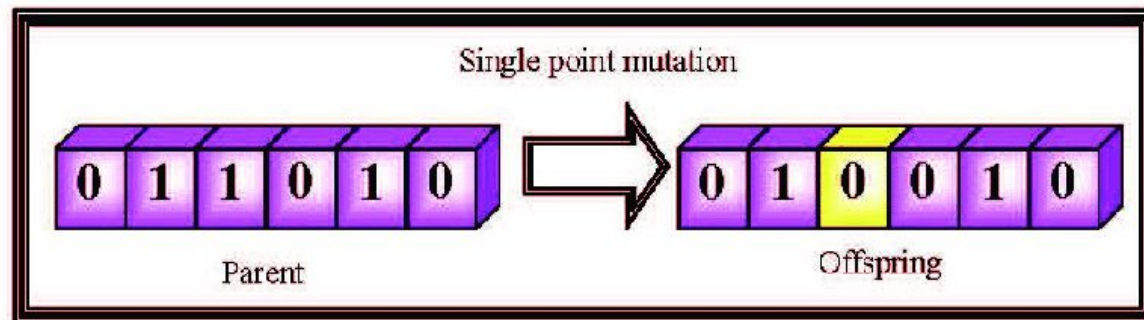
- PMX Crossover

Parent 1: 8 4 7 3 6 2 5 1 9 0  
Parent 2: 0 1 2 3 4 5 6 7 8 9  
Child 1: \_ \_ 4 3 6 2 5 1 \_ \_

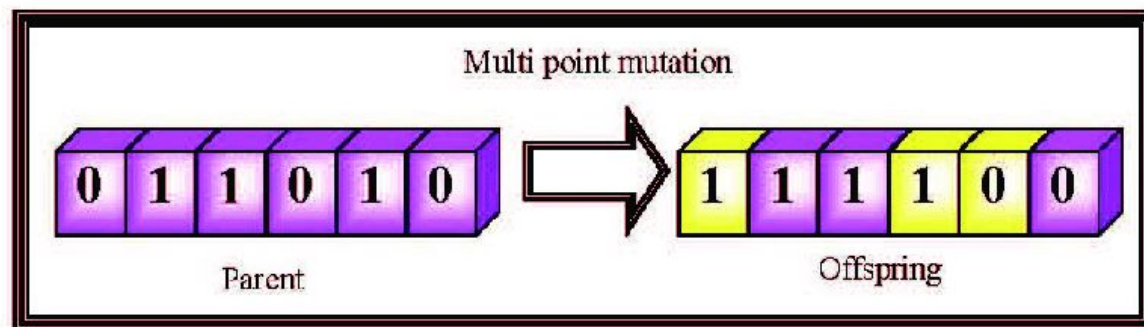
1. Chọn ngẫu nhiên 2 điểm cắt. Đoạn nằm giữa 2 điểm cắt cha 2 đem xuống con 1
2. Đối sánh các phần tử trong khoảng cắt:
  - a. 4 là vị trí đầu tiên trong khoảng cắt cha 2 không xuất hiện trong cha 1
  - b. đối sánh lên ta có phần tử 6 trong cha 1. Tìm giá trị 6 trong cha 2. Giá trị 6 này nằm ở vị trí 6 -> trong khoảng cắt -> tiếp tục lặp
  - c. Đối sánh lên 5 ở cha 1 -> giá trị 5 trong cha 2 ở vị trí 5 -> trong khoảng cắt -> tiếp tục lặp
  - d. đối đánh lên 2 ở cha 1 -> giá trị 2 trong cha 2 ở vị trí 2 -> vị trí ngoài khoảng cắt -> điền 4 vào vị trí này
3. Lặp lại cho đến khi điền hết con 1

# Toán tử đột biến - Mã hóa nhị phân

- Đột biến một điểm
- Đột biến nhiều điểm



(a)



(b)

# Toán tử đột biến - Mã hóa hoán vị



Insert Mutation for permutations



Swap mutation for permutations



Inversion mutation for permutations



# Điều kiện dừng

---

- Thuật toán dừng khi:
  - Đã tìm được lời giải tối ưu (làm sao biết đâu là lời giải tối ưu)
  - Lời giải tìm được đã tốt hơn ngưỡng cho trước
  - Thuật toán đã hội tụ: Không thể cải thiện được lời giải sau một hoặc một số thế hệ


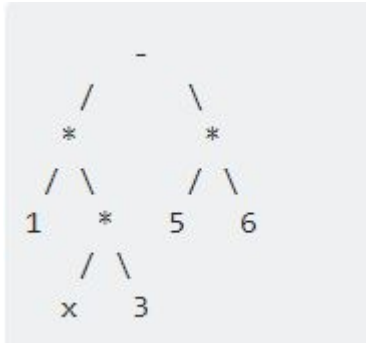


# Genetic Programming

---

- Có thể coi là một trường hợp đặc biệt của Genetic Algorithm
- Cũng giống như GA gồm các bước như mã hóa, chọn lọc, lai ghép và đột biến

# Genetic Programming vs Genetic Algorithm

Genetic Algorithm	Genetic Programming
<p>Biểu diễn mã hóa dưới dạng một chuỗi các hoạt động hoặc giá trị, thường là string hoặc list.</p> <p>-&gt; Có thể là một chuỗi hoặc một list dữ liệu thô ( mặc dù đã được mã hóa)</p>	<p>Biểu diễn mã hóa dưới dạng có cấu trúc của các hoạt động hoặc giá trị, thường là nested data.</p> <p>-&gt; Có thể là một chương trình máy tính (computer program)</p>
	





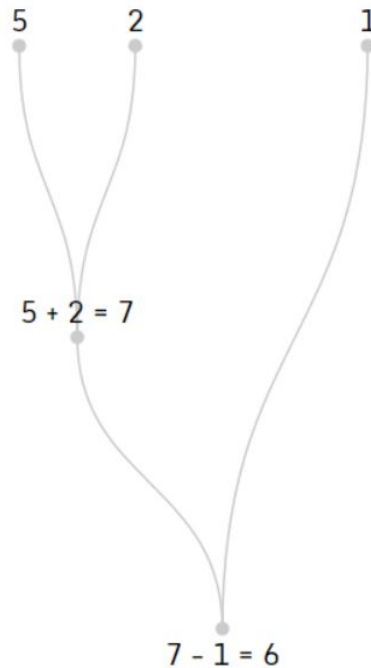
# Genetic Programming vs Genetic Algorithm

Genetic Algorithm	Genetic Programming
Nhiệm sắc thể của GA có độ dài cố định -> không gian tìm kiếm đã xác định được cận trên	Nhiệm sắc thể của GP có độ dài thay đổi. Có thể thoải mái hơn nhưng độ phức tạp cũng có thể tăng nhanh hơn.
Bởi vì không có cấu trúc nên cá thể sinh ra có thể rơi vào trường hợp cá thể không hợp lệ.	Hiếm khi rơi vào trường hợp cá thể không hợp lệ, ta có thể hủy bỏ chúng ngay trong quá trình lai ghép.



# Genetic Programming Application

VD: Cho kết quả 6, sinh một biểu thức gồm các toán tử +, -, \*, / bằng 6. Ta có thể thêm các ràng buộc cho bài toán trên như phải sử dụng số nào và không được sử dụng số nào.





# Genetic Programming Application

---

VD: Phân tích một số thành tích của hai thừa số nguyên tố.  
Bài toán quan trọng trong bảo mật.

$$2627 = 37 * 71$$

---

Thank you for listening

---