

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HCM**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA KHOA HỌC MÁY TÍNH**

**CS410.L11.KHTN**

**MẠNG NEURAL VÀ THUẬT GIẢI DI TRUYỀN**

BÁO CÁO BÀI TẬP 1

ONE MAX PROBLEM

Nguyễn Văn Tiến

18521489 – KHTN2018

# I. CÀI ĐẶT THÍ NGHIỆM

## 1. Cấu trúc mã nguồn

./

+---figure

| +---1MAX

| | evaluations.png

| | MRPS.png

| |

| +---TRAP5

| | evaluations.png

| | MRPS.png

| |

+---hypothesis

| +---1MAX

| | +---1X

| | | 010.npy

| | | 020.npy

| | | 040.npy

| | | 080.npy

| | | 160.npy

| | |

| | \---UX

| | 010.npy

| | 020.npy

| | 040.npy

| | 080.npy

| | 160.npy

| |

| \---TRAP5

| +---1X

| | 010.npy

| | 020.npy

| | 040.npy

| | 080.npy

| | 160.npy

| |

| \---UX

| 010.npy

| 020.npy

| 040.npy

| 080.npy

| 160.npy

|

+---notebook

| OneMaxAndTrap5.ipynb

| sGA\_with\_POPOP\_solve\_OneMax\_function.ipynb

|

\---src

config.py

main.py

output.doc

sGA\_onemax.py

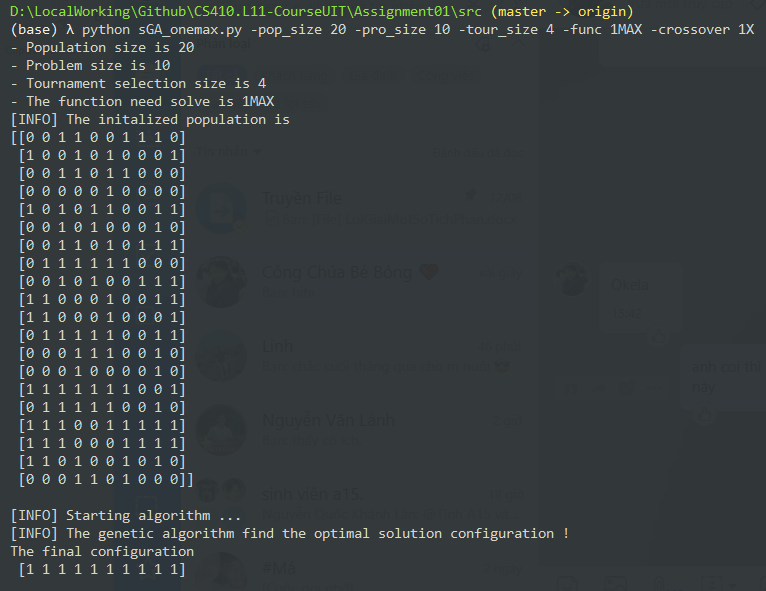
utils.py

visualize.py

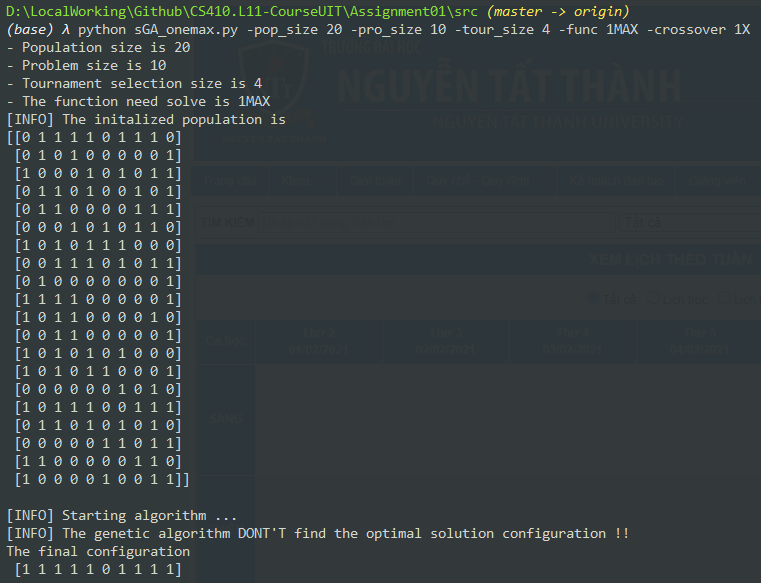
## 2. Mô tả các bước cài đặt thuật toán trong mã nguồn

* Thí nghiệm cài đặt mô hình POPOP tìm lời giải tối ưu hàm OneMax và Trap5 với hai phép lai tạo (crossover) là 1X và UX.
* Thí nghiệm được cài đặt theo hướng tiếp cận ghép các module (hàm) lại với nhau thành module lớn hơn, cụ thể như sau:
  + Cài đặt các hàm cần thiết và lưu trữ trong file ***utils.py***:
    - initialize\_population(N: int, l: int, distribution: dict)   
      🡪 Khởi tạo quần thể trước mỗi lần chạy thuật toán tiến hóa.
    - evaluation(individual: np.ndarray, \_type="1MAX", k=5) 🡪 Đánh giá độ thích nghi của 1 cá thể.
    - average\_evaluation(population: np.ndarray, optimized\_function:str)  
      🡪 Tính giá trị thích nghi trung bình của quần thể.
    - crossover(population: np.ndarray, crossover\_way='1X', threshold=None)  
      🡪 Thực hiện phép lai tạo từ quần thể trước đó tạo ra các cá thể con.
    - pop\_pool(population: np.ndarray, offspring:np.ndarray)  
      🡪 Gom nhóm cá thể con và cá thể cha mẹ.
    - tournament\_selection(population: np.ndarray, tournament\_size:int, optimized\_function:str)  
      🡪 Thực hiện quá trình chọn lọc cá thể bằng phương pháp tournament selection.
    - check\_convergence(population: np.ndarray)  
      🡪 Kiểm tra quần thể hiện tại đã hội tụ hay chưa.
  + Thuật toán tiến hóa (Genetic algorithm) theo mô hình POPOP được cài đặt trong hàm ***genetic\_algorithm()*** trong file ***sGA\_onemax.py.***🡪 Câu lệnh thực thi:

python sGA\_onemax.py -pop\_size 20 -pro\_size 10 -tour\_size 4 -func 1MAX -crossover 1X

*Trường hợp thuật toán tìm thấy lời giải tối ưu.*

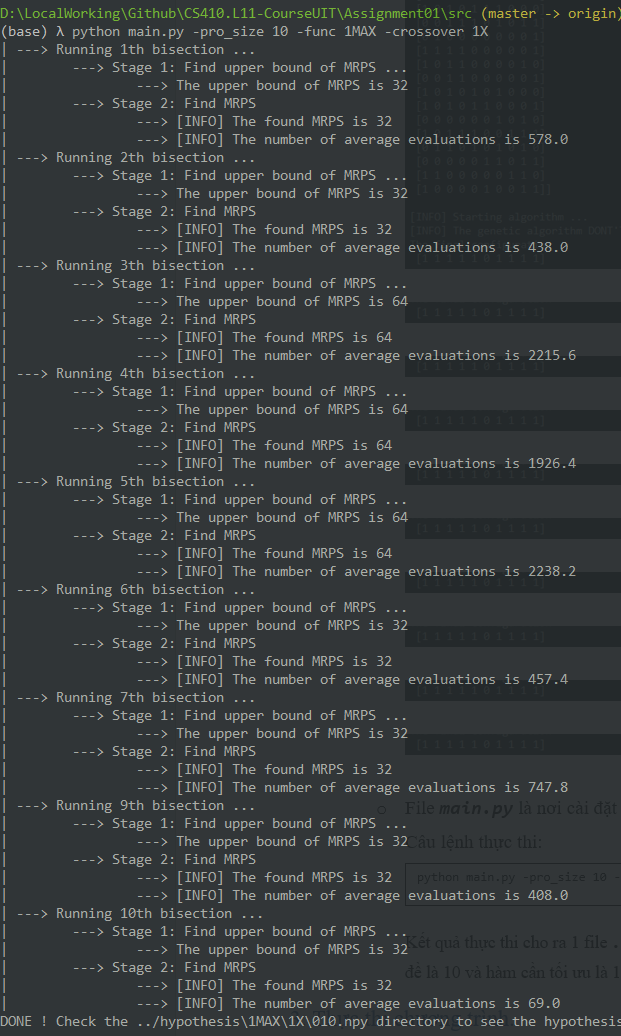
*Trường hợp thuật toán* ***không*** *tìm thấy lời giải tối ưu*



* + File ***main.py*** là nơi cài đặt các phép thí nghiệm dựa theo mô tả đề bài.  
    🡪 Câu lệnh thực thi:

python main.py -pro\_size 10 -func 1MAX -crossover 1X

*Kết quả thực thi cho ra 1 file* ***.npy*** *lưu trữ kết quả chạy của bisection với thước vấn đề là 10 và hàm cần tối ưu là 1MAX, kĩ thuật lai ghép là 1X.*

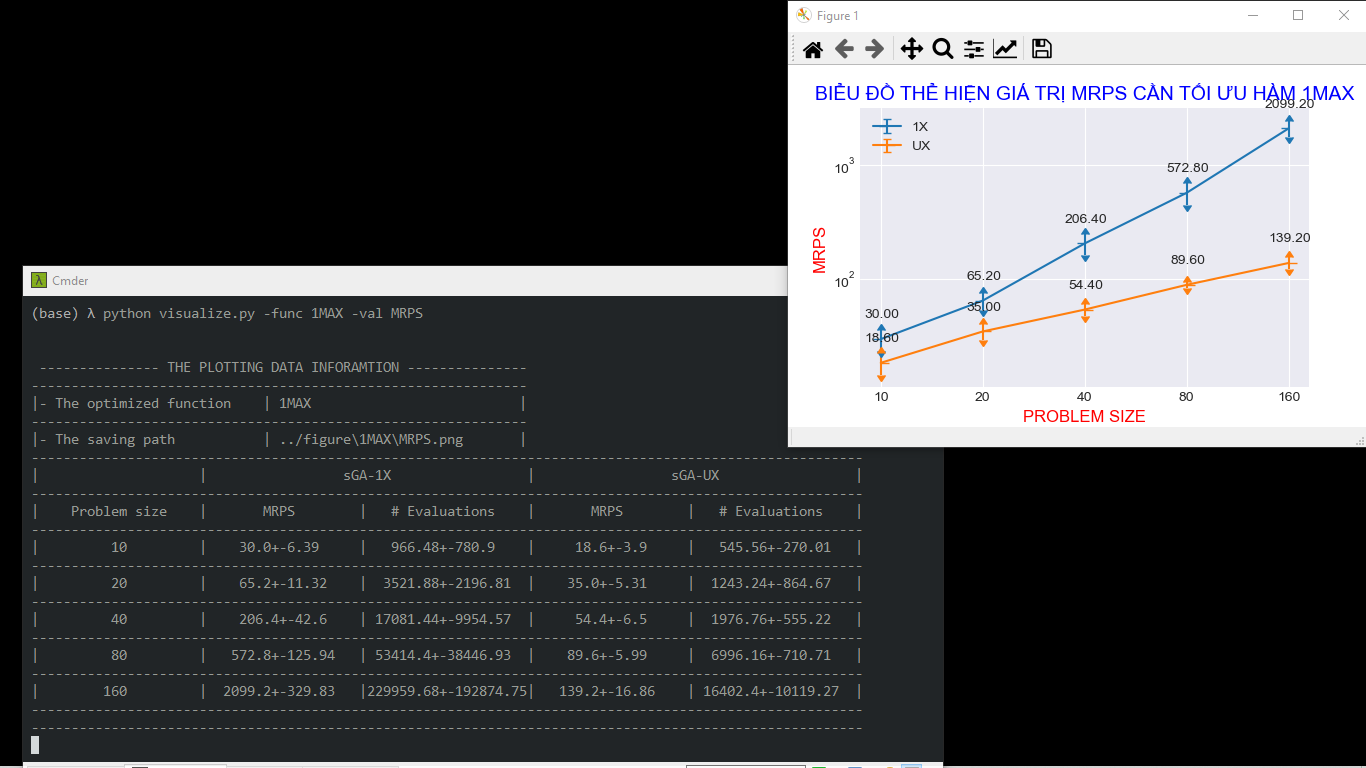


Chúng ta có 2 hàm cần thử nghiệm là OneMax và Trap5, với mỗi hàm ta chạy kết quả thống kê ứng phép lai tạo 1X hoặc UX với các kích thước vấn đề là .

* + File ***visualize.py*** dùng để trực quan hóa kết quả thí nghiệm được lưu lại sau các lần chạy file ***main.py***.  
    🡪 Câu lệnh thực thi:

python visualize.py -func 1MAX -val MRPS

Kết quả thực thi câu lệnh là biểu đồ được biểu diễn giá trị **MRPS** khi kích thước vấn đề (**problem size**) thay đổi, thu được khi chạy thí hiệm cho bài toán tối ưu hàm OneMax.



# II. KẾT QUẢ THÍ NGHIỆM

## 1. Thí nghiệm giải bài toán tối ưu hàm One Max

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **sGA-1X** | | **sGA-UX** | |
| **Problem size** | **MRPS** | **# Evaluations** | **MRPS** | **# Evaluations** |
| **10** |  |  |  |  |
| **20** |  |  |  |  |
| **40** |  |  |  |  |
| **80** |  |  |  |  |
| **160** |  |  |  |  |

**Bảng số liệu 1.** Kết quả thí nghiệm thuật toán tiến hóa tối ưu hàm OneMax theo mô hình POPOP sử dụng phép lai tạo 1X và UX.

Chart, line chart

Description automatically generated

**Hình 1.** Biểu đồ giá trị **MRPS** tối ưu hàm **OneMax** sử dụng phép lai **1X** và **UX**.

Chart, line chart

Description automatically generated

**Hình 2.** Biểu đồ giá trị **# Evaluations** tối ưu hàm **OneMax** sử dụng phép lai **1X** và **UX**

## 2. Thí nghiệm giải bài toán tối ưu hàm Trap5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **sGA-1X** | | **sGA-UX** | |
| **Problem size** | **MRPS** | **# Evaluations** | **MRPS** | **# Evaluations** |
| **10** |  |  |  |  |
| **20** |  |  |  |  |
| **40** |  |  |  |  |
| **80** |  |  |  |  |
| **160** |  |  |  |  |

**Bảng số liệu 2.** Kết quả thí nghiệm thuật toán tiến hóa theo mô hình POPOP tối ưu hàm **Trap5** sử dụng phép lai tạo 1X và UX.

**Hình 3.** Biểu đồ giá trị MRPS tối ưu hàmTrap5 sử sụng phép lai 1X và UX

**Hình 4.** Biểu đồ giá trị # Evaluations tối ưu hàmTrap5 sử sụng phép lai 1X và UX

# III. ĐÁNH GIÁ

* Phép lai ghép **1X** áp dụng cho bài toán tối ưu hàm **OneMax** có chi phí cao hơn phép lai ghép **UX** vì …
* Phép lai ghép **1X** áp dụng cho bài toán tối ưu hàm **Trap5** có chi phí thấp hơn phép lai ghép **UX** vì …
* ABC XYZ

***--- Hết ---***