

TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA Công Nghệ Thông Tin
BỘ MÔN: Công Nghệ Phần Mềm
ĐỀ THI VÀ BÀI LÀM

Tên học phần: Trí tuệ nhân tạo

Mã học phần: Hình thức thi: *Tự luận có giám sát*

Đề số: **0003** Thời gian làm bài: 70 phút (*không kể thời gian chép/phát đề*)

Được sử dụng tài liệu khi làm bài.

Họ tên: ...Nguyễn Quý Triều.....**Lớp:**...19TCLC_DT3.....**MSSV:**....102190145.....

Sinh viên làm bài trực tiếp trên tệp này, lưu tệp với định dạng MSSV_HọTên.pdf và nộp bài thông qua MSTeam.

Câu 1 (2 điểm): Cho không gian Oxy với các điểm có tọa độ tương ứng (3, 12), (6, 9), (4.5, 12), (12, 3), (15, 6), (12, 15), (9, 18) và (9, 12).

a) (1 điểm) Mô tả thuật toán hoặc hàm thực thi thuật toán *k*-means

Trả lời: viết mô tả thuật toán hoặc dán code vào bên dưới

Input: Dữ liệu *X* và số lượng cluster cần tìm *K*.

Output: Các center *M* và label vector cho từng điểm dữ liệu *Y*.

1. Chọn *K* điểm bất kỳ làm các center ban đầu

2. Phân mỗi điểm dữ liệu vào cluster có center gần nó nhất.

3. Nếu việc gán dữ liệu vào từng cluster ở bước 2 không thay đổi so với vòng lặp trước nó thì ta dừng thuật toán.

4. Cập nhật center cho từng cluster bằng cách lấy trung bình cộng của tất cả các điểm dữ liệu đã được gán vào cluster đó sau bước 2.

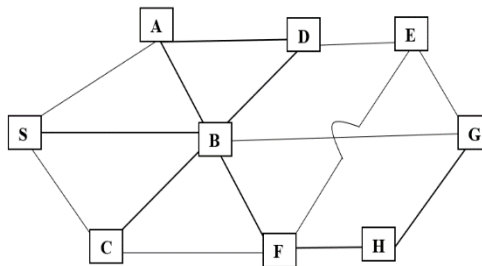
5. Quay lại bước 2.

b) (1 điểm) Nếu sử dụng thuật toán *k*-means với *k* = 3 thì kết quả phân nhóm sẽ như thế nào? (các điểm thuộc mỗi nhóm, trọng tâm của mỗi nhóm).

Trả lời: viết câu trả lời vào bên dưới

Chia làm 3 cụm

Câu 2 (5 điểm): Cho đồ thị vô hướng $G = (V, E)$ như hình vẽ với *V* là tập đỉnh và *E* là tập cạnh.



- a) (2 điểm) Hãy viết đoạn code biểu diễn đồ thị trên bằng cách khởi tạo tập đỉnh V và tập cạnh E.
(Ví dụ: V = ["S", "A", "B"], E = [("S", "A"), ("S", "B")])

Trả lời: Dán code vào bên dưới

```
V = ["S", "A", "B", "C", "D", "E", "F", "G", "H"]
E = [("S", "A"), ("A", "S"), ("S", "B"), ("B", "S"), ("S", "C"), ("C", "S"), ("A", "B"), ("B", "A"), ("A", "D"), ("D", "A"), ("B", "D"), ("D", "B"), ("B", "G"), ("G", "B"), ("B", "F"), ("F", "B"), ("C", "F"), ("F", "C"), ("C", "B"), ("B", "C"), ("D", "E"), ("E", "D"), ("F", "E"), ("E", "F"), ("F", "H"), ("H", "F"), ("E", "G"), ("G", "E"), ("H", "G"), ("G", "H")]
```

- b) (3 điểm) Hãy viết chương trình sử dụng thuật toán **tìm kiếm theo chiều rộng (BFS)** để tìm đường đi từ đỉnh "S" đến đỉnh "G" trong đồ thị được biểu diễn ở câu a). Trong chương trình, hãy in ra thứ tự đỉnh khám phá trong quá trình tìm kiếm. Nếu không tìm thấy thì in "Không tìm thấy đường đi"

Trả lời: Dán code vào bên dưới

```
V = ["S", "A", "B", "C", "D", "E", "F", "G", "H"]
E = [("S", "A"), ("A", "S"), ("S", "B"), ("B", "S"), ("S", "C"), ("C", "S"), ("A", "B"), ("B", "A"), ("A", "D"), ("D", "A"), ("B", "D"), ("D", "B"), ("B", "G"), ("G", "B"), ("B", "F"), ("F", "B"), ("C", "F"), ("F", "C"), ("C", "B"), ("B", "C"), ("D", "E"), ("E", "D"), ("F", "E"), ("E", "F"), ("F", "H"), ("H", "F"), ("E", "G"), ("G", "E"), ("H", "G"), ("G", "H")]
```

```
Start = V[0]
Open = [Start]
Goal = "G"
close = []
dinhcha = {}
Ok = False
count = 0
while len(Open) > 0:
    count += 1
    O = Open.pop(0)
```

```

close.append(0)
if O == Goal:
    Ok = True
    break
for child1, child in E:
    if child1 == O:
        if child in Open or child in close:
            continue
        Open.append(child)
        close.append(child)
        dinhcha[child] = O
    elif child == O:
        if child1 in Open or child1 in close:
            continue
        Open.append(child1)
        close.append(child1)
        dinhcha[child1] = O
truyvet = [O]
if Ok== False:
    print("Khong tim thay duong di")
if Ok:
    while dinhcha.get(O) !=None:
        O = dinhcha.get(O)
        truyvet.append(O)
truyvet.reverse()
print("Đường đi từ đỉnh S đến đỉnh G là:", *truyvet)

```

Trả lời: Dán kết quả thực thi vào bên dưới:

```

Đường đi từ đỉnh S đến đỉnh G là: S B G
PS D:\Trường bách khoa\Học Kỳ 5\Trí tuệ nhân tạo\cuối kì> py bai2.py
Đường đi từ đỉnh S đến đỉnh G là: S B G
PS D:\Trường bách khoa\Học Kỳ 5\Trí tuệ nhân tạo\cuối kì>

```

Câu 3(3 điểm): Cho hàm $f(x) = \left(e^{-x} - \frac{4}{e^{-2x}}\right)^2$

a) (1 điểm) Khai triển đạo hàm cấp 1 của $f(x)$

Trả lời: Khai triển và kết quả đạo hàm ở bên dưới

$$\begin{aligned}
 d/dx(-4e^{2x}+e^{-x}) &= (-8e^{2x}-e^{-x})(-8e^{2x}+2e^{-x}) \\
 &= 64e^{4x}-8e^x-2e^{-2x}
 \end{aligned}$$

b) (2 điểm) Viết chương trình tính giá trị nhỏ nhất của $f(x)$ sử dụng thuật toán Gradient Descent

Trả lời: viết câu trả lời vào bên dưới

```
import numpy as np
import math
def grad(x):
    return 64*math.exp(4*x)-8*math.exp(x)-2*math.exp(-2*x)
def cost(x):
    return (math.exp(-x)- 4/math.exp(-2*x))*(math.exp(-x)- 4/math.exp(-2*x))
def myGD1(x0,alpha=0.0001, gra = 0.00001, loop = 10000):
    x = [x0]
    for it in range(loop):
        x_new = x[-1] - alpha*grad(x[-1])
        if abs(grad(x_new)) < gra:
            break
        x.append(x_new)
    return (x, it)
if __name__ == '__main__':
    (x3, it3) = myGD1(-1,0.0001)
    print('giá trị x tìm được = %f, giá trị fx = %f, tìm ra sau %d bước lặp'%(x3[-1], cost(x3[-1]), it3))
```

Trả lời: Dán kết quả thực thi vào bên dưới

```
PS D:\Trường bách khoa\Học Kỳ 5\Trí tuệ nhân tạo\cuối kì> py bai3.py
giá trị x tìm được = -0.462098, giá trị fx = 0.000000, tìm ra sau 3384 bước lặp
PS D:\Trường bách khoa\Học Kỳ 5\Trí tuệ nhân tạo\cuối kì>
```

GIẢNG VIÊN BIÊN SOẠN ĐỀ THI

Đà Nẵng, ngày 26 tháng 11 năm 2021
TRƯỞNG BỘ MÔN
(đã duyệt)