

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: Bùi Kim Phát

Lớp: 19TCLC\_DT3

MSSV: 102190132

**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

- Khởi tạo  $k$  điểm dữ liệu trong bộ dữ liệu và tạm thời coi nó là các tâm của các cụm dữ liệu. Khởi tạo cụm ban đầu mà tất cả điểm dữ liệu thuộc về.
- Với mỗi điểm trong bộ dữ liệu, tâm cụm sẽ được xác định là 1 trong  $k$  tâm cụm gần nó nhất mà mình đang xác định.
- Sau khi tất cả các điểm dữ liệu đã có tâm cụm, tính toán lại vị trí mới của các tâm cụm đó để tâm cụm nằm giữa cụm.
- Lặp lại việc xác định cụm của từng điểm dữ liệu và xác định tâm cụm mới cho đến khi vị trí của các tâm cụm không đổi (hoặc chỉ đổi 1 khoảng không đáng kể).

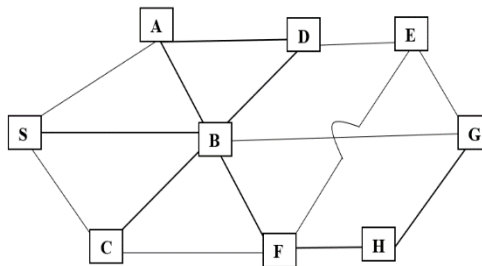
b) (1 điểm) Nếu sử dụng thuật toán  $k$ -means với  $k = 3$  thì kết quả phân nhóm sẽ như thế nào? (các điểm thuộc mỗi nhóm, trọng tâm của mỗi nhóm).

**# Trả lời:** viết câu trả lời vào bên dưới

Với  $k = 3$

- Nhóm 0 gồm các điểm:  $(3, 12)$ ,  $(6, 9)$ ,  $(4.5, 12)$ ; có trọng tâm là:  $(4.5, 11)$
- Nhóm 1 gồm các điểm:  $(12, 3)$ ,  $(15, 6)$ ; có trọng tâm là:  $(13.5, 4.5)$
- Nhóm 2 gồm các điểm:  $(12, 15)$ ,  $(9, 18)$ ,  $(9, 12)$ ; có trọng tâm là:  $(10, 15)$

**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', 'H', 'G']
E = [[0,1,1,1,0,0,0,0,0],
      [1,0,1,0,1,0,0,0,0],
      [1,1,0,1,1,0,1,0,1],
      [1,0,1,0,0,0,1,0,0],
      [0,1,1,0,1,0,0,0,0],
      [0,0,0,0,1,0,1,0,1],
      [0,0,1,1,0,1,0,1,0],
      [0,0,0,0,0,0,1,0,1],
      [0,0,1,0,0,1,0,1,0]]
```

- 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', 'H', 'G'];
E = [[0,1,1,1,0,0,0,0,0],
      [1,0,1,0,1,0,0,0,0],
      [1,1,0,1,1,0,1,0,1],
      [1,0,1,0,0,0,1,0,0],
      [0,1,1,0,1,0,0,0,0],
      [0,0,0,0,1,0,1,0,1],
      [0,0,1,1,0,1,0,1,0],
      [0,0,0,0,0,0,1,0,1],
      [0,0,1,0,0,1,0,1,0]]

start = 0
goal = 8
open = [start]
close = []
OK = False;
dinhcha = {}
while len(open)>=1 :
    O = open.pop(0)
    close.append(O)
    if O == goal:
        OK = True
        break
    possible = []
    for y in range(0,9):
```

```

if E[0][y] == 0:
    continue
if y in close or y in open:
    continue
possible.append(y)
nexts = []
for i in possible:
    nexts.append(V[i])
    child = i
    open.append(child)
    dinhcha[child] = 0
print("Tai " + V[0])
print("Cac dinh tiep theo: ",nexts)
if OK == False:
    print("Khong tim thay duong di")
else:
    truyvet = [V[0]]
    while dinhcha.get(0) !=None:
        O = dinhcha.get(0)
        truyvet.append(V[O])
    truyvet.reverse()
    print("Duong di: ")
    print(*truyvet)

```

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

```

Tai S
Cac dinh tiep theo:  ['A', 'B', 'C']
Tai A
Cac dinh tiep theo:  ['D']
Tai B
Cac dinh tiep theo:  ['F', 'G']
Tai C
Cac dinh tiep theo:  []
Tai D
Cac dinh tiep theo:  []
Tai F
Cac dinh tiep theo:  ['E', 'H']
Duong di:
S B G

```

**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

$$-2e^{-2x} - 8e^x + 64e^{4x}$$

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 math
def f(x):
    return math.exp(-2*x) - 8*math.exp(x) + 16*math.exp(4*x)

def df(x):
    return -2*math.exp(-2*x)-8*math.exp(x)+64*math.exp(4*x)

lr = 0.2
x = 0
for i in range(1000):
    while abs(df(x)) > 0.001:
        x = x - lr*df(x)
print("x = " + str(x))
print("y = " + str(f(x)))

# Trả lời: Dán kết quả thực thi vào bên dưới
x = 0,462
y = 0
```

**GIẢNG VIÊN BIÊN SOẠN ĐỀ THI**

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