I. Lý thuyết

1. BFS

Đẩy từ phải vào, **không lặp lại**.

2. UCS

- BFS có trọng số, chọn nút có giá thành nhỏ nhất để mở rộng
- **Giá thành cập nhật theo giá trị nhỏ nhất, sẽ được đưa lại tập biên nếu đã mở rộng hoặc cập nhật thay nút cũ có giá thành kém hơn nếu đang trong tập biên.**

3. DFS

Đẩy từ trái vào, **có thể đẩy lặp nếu chưa được mở rộng**.

4. IDS

- DFS sâu dần (tăng dần giới hạn độ sâu), **có thể đẩy lặp kể cả khi đã mở rộng hay đã trong tập biên**.
- Nếu depth < c, đẩy từ trái vào

5. Tham lam

Mở rộng theo giá trị heuristic (h(n)) nhỏ nhất, **không lặp lại**

6. A*

- Công thức:

$$f(m) = g(m) + h(m) \text{ và } g(m) = g(n) + c(n, m)$$

Với: g(m): giá thành đường đi từ điểm xuất phát tới nút m

h(m): hàm heuristic ước lượng giá thành đường đi từ m tới đích

f(m): ước lượng giá thành đường đi từ điểm xuất phát qua m tới đích

- Mở rộng theo giá thành ước lượng nhỏ nhất (f(m)).
- **Giá thành ước lượng cập nhật theo giá thành nhỏ nhất, sẽ được đưa lại tập biên nếu đã mở rộng hoặc cập nhật thay nút cũ có giá thành kém hơn nếu đang trong tập biên.**
- Hàm heuristic (h(n)) là 1 hàm chấp nhận được nếu: Mọi nút n thì h(n) <= h*(n) với h*(n) là giá thành thực nhỏ nhất để đi từ n đến đích.
- Nếu h là hàm heuristic không thể chấp nhận, kết quả tìm đường đi từ thuật toán A* có thể không phải đường đi tối ưu.

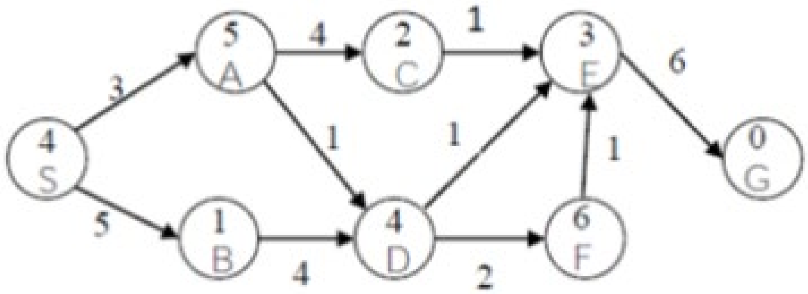
- Nếu h1(n) và h2(n) là 2 hàm heuristic chấp nhận được thoả mãn h1(n) <= h2(n) với mọi nút n thì h2 trội hơn (tốt hơn) h1.

7. IDA* (A* sâu dần)

- **Đẩy từ trái vào, có thể đẩy lặp kể cả khi đã mở rộng hay đã trong tập biên**
- Xét thêm giá trị i tăng dần, nếu f(m) <= i thì thêm vào tập biên O (mở rộng)
- Với trường hợp giá trị $\alpha = 1$ thì IDA* luôn tìm được đường đi tối ưu.

II. Bài tập

1. Sử dụng 7 thuật toán tìm kiếm tìm đường đi tối ưu từ S đến G (bởi bài này cho rõ giá thành đường đi nên kể cả các thuật toán không sử dụng đến vẫn phải kết luận chi phí):



BL

- BFS:

STT	Nút được mở rộng	Tập biên O
0		S
1	S	A_S, B_S
2	A_S	B_S, C_A, D_A
3	B_S	C_A, D_A
4	C_A	D_A, E_C
5	D_A	E_C, F_D
6	E_C	F_D, G_E
7	F_D	G_E
8	G_E	Đích

Đường đi: G <- E <- C <- A <- S với chi phí 14

- DFS:

STT	Nút được mở rộng	Tập biên O
0		S
1	S	A_S, B_S
2	A_S	C_A, D_A, B_S
3	C_A	E_C, D_A, B_S
4	E_C	G_E, D_A, B_S
5	G_E	Đích

Đường đi: G <- E <- C <- A <- S với chi phí 14

- UCS:

STT	Nút được mở rộng	Tập biên O
0		S(0)
1	S	$A_S(3), B_S(5)$
2	A_S	$B_S(5), C_A(7), D_A(4)$
3	D_A	$B_S(5), C_A(7), E_D(5), F_D(6)$
4	B_S	$C_A(7), E_D(5), F_D(6)$
5	E_D	$C_A(7), F_D(6), G_E(11)$
6	F_D	$C_A(7), G_E(11)$
7	C_A	$G_E(11)$
8	G_E	Đích

Đường đi: G <- E <- D <- A <- S với chi phí 11

- IDS:

Nút được mở rộng	Tập biên O
C = 0	
	S
S	$\emptyset$
C = 1	
	S
S	A_S, B_S
A_S	B_S
B_S	$\emptyset$
C = 2	
	S
S	A_S, B_S
A_S	C_A, D_A, B_S
C_A	D_A, B_S
D_A	B_S
B_S	D_B
D_B	$\emptyset$
C = 3	
	S
S	A_S, B_S
A_S	C_A, D_A, B_S
C_A	E_C, D_A, B_S
E_C	D_A, B_S
D_A	E_D, F_D, B_S
E_D	F_D, B_S
F_D	B_S
B_S	D_B
D_B	E_D, F_D

E_D	F_D
F_D	$\emptyset$
C = 4	
	S
S	A_S, B_S
A_S	C_A, D_A, B_S
C_A	E_C, D_A, B_S
E_C	G_E, D_A, B_S
G_E	Đích

Đường đi: G <- E <- C <- A <- S với chi phí 14

- Tham lam:

STT	Nút được mở rộng	Tập biên O
0		S(4)
1	S	$A_S(5), B_S(1)$
2	B_S	$A_S(5), D_B(4)$
3	D_B	$A_S(5), E_D(3), F_D(6)$
4	E_D	$A_S(5), F_D(6), G_E(0)$
5	G_E	Đích

Đường đi: G <- E <- D <- B <- S với chi phí 16

- A*:

STT	Nút được mở rộng	Tập biên O
0		S(0 + 4)
1	S	$A_S(3 + 5), B_S(5 + 1)$
2	B_S	$A_S(3 + 5), D_B(9 + 4)$
3	A_S	$C_A(7 + 2), D_A(4 + 4)$
4	D_A	$C_A(7 + 2), E_D(5 + 3), F_D(6 + 6)$
5	E_D	$C_A(7 + 2), F_D(6 + 6), G_E(11 + 0)$
6	C_A	$F_D(6 + 6), G_E(11 + 0)$
7	G_E	Đích

Đường đi: G <- E <- D <- A <- S với chi phí 11

- IDA* với $\alpha = 8$:

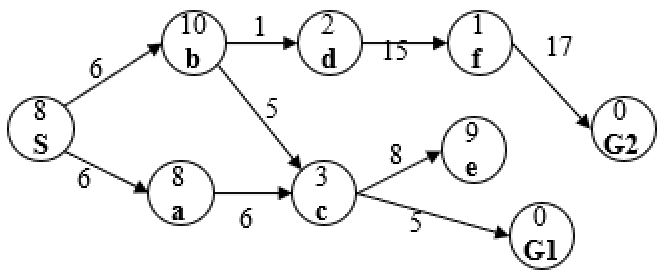
Nút được mở rộng	Tập biên O
i = 0	
	S
S	$\emptyset$
i = 8	
	S
S	$A_S(3 + 5), B_S(5 + 1)$
A_S	$D_A(4 + 4), B_S(5 + 1)$
D_A	$E_D(5 + 3), B_S(5 + 1)$

E_D	$B_S(5+1)$
B_S	$\emptyset$
i = 16	
	S(4)
S	$A_S(3+5), B_S(5+1)$
A_S	$C_A(7+2), D_A(4+4), B_S(5+1)$
C_A	$E_C(8+3), D_A(4+4), B_S(5+1)$
E_C	$G_E(14+0), D_A(4+4), B_S(5+1)$
G_E	Đích

Đường đi: G <- E <- C <- A <- S với chi phí 14.

Ta có thể thử giá trị α khác và có thể nhận ra rằng đường đi này không phải đường đi tối ưu, đường đi tối ưu có chi phí 11 là G <- E <- D <- A <- S. Lý do có trường hợp IDA* không tìm được đường đi tối ưu là do giá trị α do ta thêm vào ngưỡng sau mỗi vòng lặp.

2. Cho đồ thị như hình vẽ, S là nút xuất phát, G1 và G2 là nút đích. Các số nằm cạnh cung là giá thành đường đi, số nằm trong vòng tròn là giá trị hàm heuristic.



a) Hàm heuristic trên hình có phải hàm chấp nhận được không ? Tại sao ?

b) Sử dụng thuật toán A* tìm đường đi từ nút xuất phát tới đích.

BL

a) Dễ thấy:

- Tại nút S, bất cứ đường đi nào đến 1 trong 2 nút đích là G1 và G2 đều có h*(S) > h(S) = 8.
- Tại nút B: h(B) = 10 <= h*(B) = 5 + 5 = 10 (nút đích chọn là G1)
- Tại nút D: h(D) = 2 <= h*(D) = 32 (nút đích chọn là G2)
- Tại nút F: h(F) = 1 <= h*(F) = 17 (nút đích chọn là G2)
- Tại nút A: h(A) = 8 <= h*(A) = 11 (nút đích chọn là G1)
- Tại nút C: h(C) = 3 <= h*(C) = 5 (nút đích chọn là G1)
- Tại nút E: không có đường đi tới bất cứ nút đích nào (coi h*(E) là vô cùng ?)

Vì mọi nút n đều có h(n) <= h*(n) nên có thể coi hàm heuristic trên hình là hàm chấp nhận được.

b) Thuật toán A*:

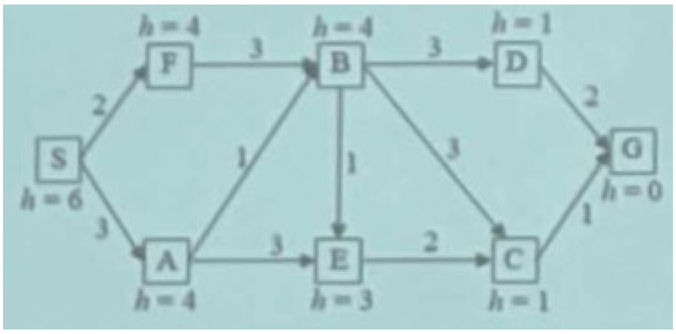
STT	Nút được mở rộng	Tập biên O
0		S(0 + 8)
1	S	$A_S(6+8), B_S(6+10)$
2	A_S	$B_S(6+10), C_A(12+3)$
3	C_A	$B_S(6+10), E_C(20+9), G1_C(17+0)$
4	B_S	$E_C(20+9), G1_C(17+0), C_B(11+3), D_B(7+2)$
5	D_B	$E_C(20+9), G1_C(17+0), C_B(11+3), F_D(22+1)$
6	C_B	$F_D(22+1), E_C(19+9), G1_C(16+0)$
7	$G1_C$	$F_D(22+1), E_C(19+9)$
8	F_D	$E_C(19+9), G2_F(39+0)$
9	E_C	$G2_F(39+0)$
10	$G2_F$	Đích

Đường đi tới đích G1: G1 <- C <- B <- S với chi phí là 16

Đường đi tới đích G2: G2 <- F <- D <- B <- S với chi phí là 39

Cách trình bày trên là tóm tắt, muốn đủ hơn nên tách thành 2 bảng cho 2 đích.

3. **Thuật toán A* được sử dụng tìm đường đi từ S tới G trên đồ thị sau. Giá thành đường đi được cho bởi các số cạnh mũi tên. Giá trị hàm heuristic h được cho cạnh mỗi nút:**



a) Điền vào bảng sau giá trị nút được mở rộng và danh sách các nút chờ cùng với giá trị hàm f sau mỗi bước.

Bước	Nút mở rộng	Danh sách nút chờ và f
1		**S(6)**
2	**S**	$\boldsymbol{A_S(7), F_S(6)}$
…		

b) Hãy làm cho hàm h trở nên không thể chấp nhận bằng cách thay đổi giá trị h tại 1 nút. Sử dụng hàm không thể chấp nhận ảnh hưởng thế nào tới A* ?

BL

a)

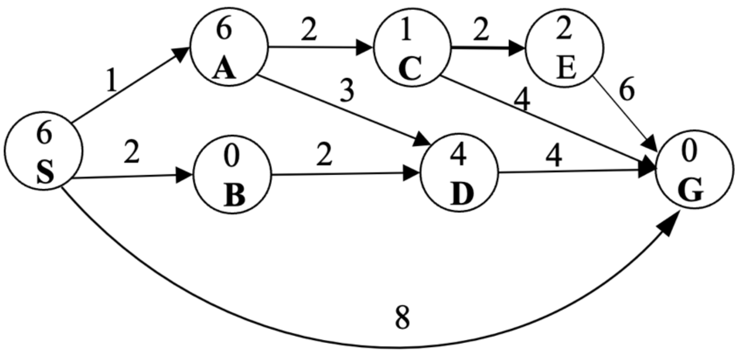
Bước	Nút mở rộng	Danh sách nút chờ và f

1		S(6)
2	S	$A_S(7), F_S(6)$
3	F_S	$A_S(7), B_F(9)$
4	A_S	$B_A(8), E_A(9)$
5	B_A	$E_B(8), C_B(8), D_B(8)$
6	C_B	$E_B(8), D_B(8), G_C(8)$
7	D_B	$E_B(8), G_C(8)$
8	E_B	$G_C(8)$
9	G_C	Đích

Đường đi: G <- C <- B <- A <- S với chi phí = 8

b) Hàm h là hàm chấp nhận được nếu: Mọi nút n thì h(n) <= h*(n) với h*(n) là giá thành thực nhỏ nhất để đi từ n đến đích. Ta có thể đổi giá trị h(C) = 2 để khiến cho hàm h trở nên không thể chấp nhận. Nếu h là hàm không thể chấp nhận, kết quả tìm đường đi từ thuật toán A* có thể không tối ưu.

4. **Cho đồ thị như trên hình vẽ, S là nút xuất phát, G là nút đích. Các số nằm cạnh cung là giá thành đường đi, số nằm trong vòng tròn là giá trị hàm heuristic:**



a) Hãy sử dụng thuật toán A* sâu dần (IDA*) với α = 4 là giá trị được thêm vào ngưỡng sau mỗi vòng lặp để tìm đường đi từ nút xuất phát tới đích. Thể hiện các giá trị: nút được mở rộng, danh sách nút biên và giá trị hàm f tại mỗi bước. Xác định đường đi do IDA* tìm được. Trong trường hợp có nhiều nút cùng tham số (độ sâu) mở rộng theo thứ tự chữ cái.

b) Đường đi tìm được ở trên có phải là đường đi tối ưu hay không? Trong trường hợp nào IDA* tìm được đường đi tối ưu?

BL

a)

Nút được mở rộng	Tập biên O
i = 0	

	S
S	∅
i = 4	
	S
S	$B_S(2+0)$
B_S	∅
i = 8	
	S
S	$A_S(1+6), B_S(2+0), G_S(8+0)$
A_S	$C_A(3+1), D_A(4+4), B_S(2+0), G_S(8+0)$
C_A	$E_C(5+2), G_C(7+0), D_A(4+4), B_S(2+0), G_S(8+0)$
E_C	$G_C(7+0), D_A(4+4), B_S(2+0), G_S(8+0)$
G_C	Đích

Đường đi: G <- C <- A <- S với chi phí = 7

b) Để kiểm tra đường đi do IDA* tìm được bên trên có tối ưu không ta có thể so sánh với đường đi tìm được bởi A*.

Để đường đi do A* tìm ra là tối ưu thì hàm heuristic phải tối ưu, thật vậy:

Tại nút E: h(E) = 2 <= h*(E) = 6

Tại nút D: h(D) = 4 <= h*(D) = 4

Tại nút C: h(C) = 1 <= h*(C) = 4

Tại nút B: h(B) = 0 <= h*(B) = 6

Tại nút A: h(A) = 6 <= h*(A) = 6

Tại nút S: h(S) = 6 <= h*(S) = 7

Với mọi nút n trên đồ thị thì h(n) <= h*(n) => hàm heuristic là hàm chấp nhận được => đường đi tìm được bởi A* là tối ưu.

Tìm đường đi bằng A*:

Bước	Nút mở rộng	Danh sách nút chờ và f
1		S(6)
2	S	$A_S(7), B_S(2), G_S(8)$
3	B_S	$A_S(7), G_S(8), D_B(8)$
4	A_S	$G_S(8), D_B(8), C_A(4)$
5	C_A	$G_S(8), D_B(8), E_C(7), G_C(7)$
6	E_C	$G_S(8), D_B(8), G_C(7)$
7	G_C	Đích

Đường đi: G <- C <- A <- S với chi phí = 7 giống đường đi do IDA* tìm được ở câu a với α = 4 nên đường đi đó là tối ưu.

Trong trường hợp $\alpha = 1$ thì IDA* luôn tìm được đường đi tối ưu.