



TRƯỜNG ĐẠI HỌC MỞ TP. HCM

Cơ hội học tập cho mọi người

Khoa Công Nghệ Thông Tin

Chương 6

CÂY BAO TRÙM TỐI TIỂU

GV: Lê Ngọc Hiếu

TP.HCM - Tháng 06 -2019



Mở đầu





Kiến thức cần thiết khi tìm hiểu về CÂY BAO TRÙM TỐI TIỂU:

- Các CTDL cơ bản, các phương pháp cơ bản về xếp thứ tự và tìm kiếm trên LIST; Cây nhị phân tìm kiếm; Đồ thị;
- Kiểu dữ liệu cơ bản, dữ liệu lưu trữ trong máy tính.
- Các kiến thức về cơ sở lập trình & kỹ thuật lập trình.

Kỹ năng cần có:

- Có thể sử dụng Visual Studio 2015
- Có thể lập trình C++

Mục tiêu dạy học

-  Hiểu được khái niệm “**Cây bao trùm**”.
-  Hiểu được khái niệm “**Cây bao trùm tối thiểu**”.
-  Hiểu và cài đặt được thuật toán **Prim**
-  Hiểu và cài đặt được thuật toán **Kruskal**

Nội dung chính

- 6.1 Cây bao trùm
- 6.2 Cây bao trùm tối thiểu
- 6.3 Thuật toán Prim
- 6.4 Thuật toán Krusal
- 6.5 Tổng kết chương
- 6.6 Bài tập chương 6

Tài liệu tham khảo

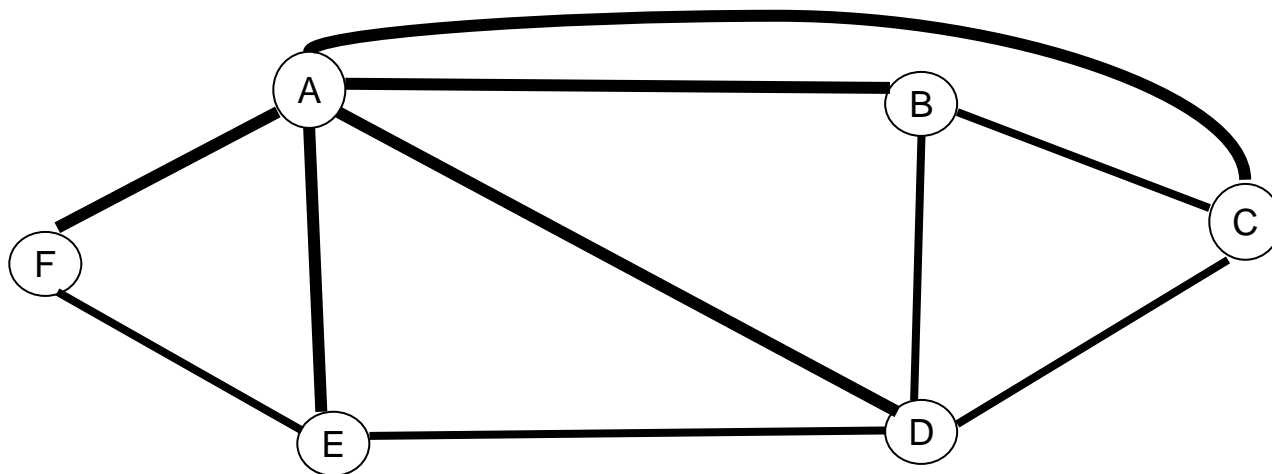


6.1

CÂY BAO TRÙM (Spanning Tree)



6.1 – CÂY BAO TRÙM

Ví dụ 6.1 Cho đồ thị $G=(V, E)$ như bên dưới



6.1 – CÂY BAO TRÙM

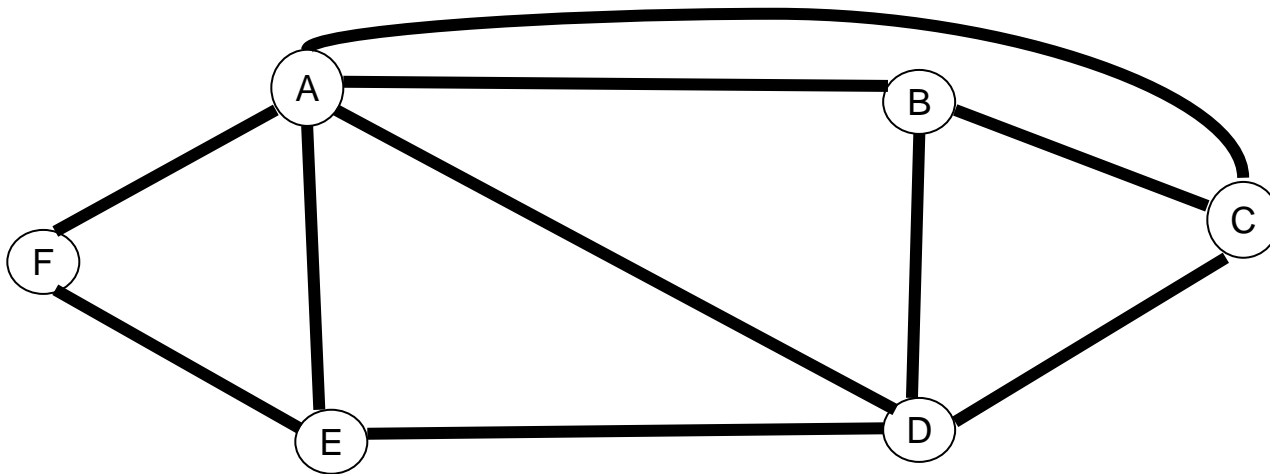
ĐỊNH NGHĨA CÂY BAO TRÙM

-  Cho đồ thị liên thông $G = (V, E)$, V là tập đỉnh, E là tập cạnh của G
-  Nếu $T = (V, E')$, với $E' \subseteq E$, và T là một cây (có nghĩa T không có chu trình)

Thì ta nói T là **cây bao trùm**

6.1 – CÂY BAO TRÙM

Xét ví dụ 6.1



$$T_2 \equiv (V, E'),$$

$$V \equiv \{A, B, C, D, E, F\};$$

$$E' \equiv \{(A,E), (B,D), (C,D), (A,B), (E,D)\} \subseteq E;$$

6.1 – CÂY BAO TRÙM

NHẬN XÉT

- 📖 Trên một đồ thị G (liên thông) có thể có nhiều cây bao trùm, gọi tập các cây khung trên G là $Sp(G)$.
- 📖 Ta có thể tìm cây bao trùm bằng các thuật giải BFS, DFS (Duyệt hết tất cả các đỉnh, mỗi lần duyệt đỉnh u nạp đỉnh vào trong T (với điều kiện $T \leftarrow \{u\}$, T không tạo ra chu trình))
- 📖 Nếu giả sử G là đồ thị có trọng số, thì *cây T cũng là cây bao trùm có trọng số.*



6.2

CÂY BAO TRÙM TỐI TIỂU

(Min Spanning Tree)

6.2 – CÂY BAO TRÙM TỐI TIỂU

ĐỊNH NGHĨA

 Cho $G = (V, E)$ là một đồ thị có trọng số,

 T là cây bao trùm tối tiểu khi:

$$w(T) = \min\{w(T) / T \in Sp(G)\}$$

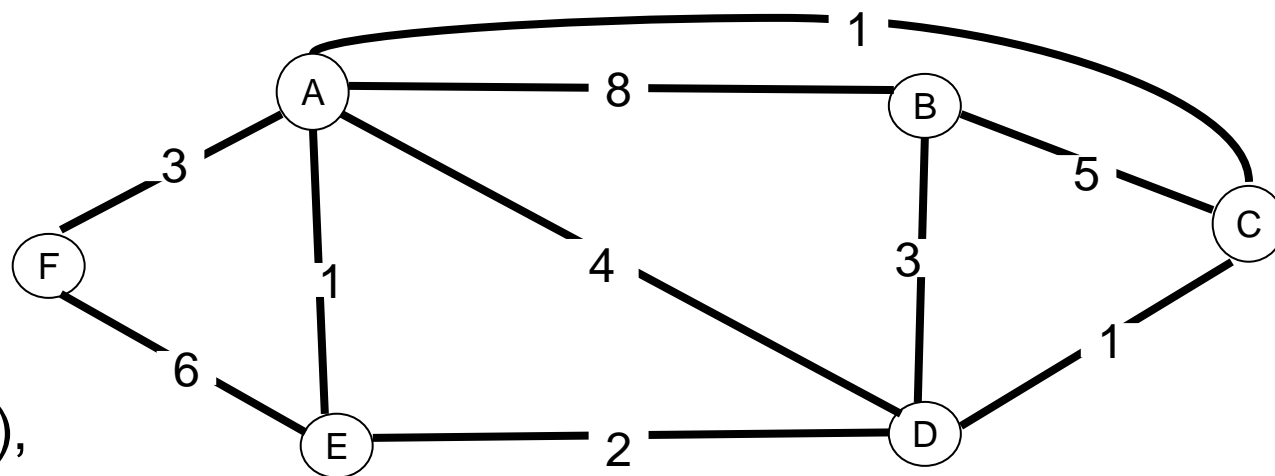
$w(T)$: tổng trọng số của các cạnh trên cây T ;

$Sp(G)$: là tập tất cả cây bao trùm trên G ;

Cây bao trùm tối tiểu là 1 cây bao trùm, có tổng trọng số là *tối tiểu* trên tập các cây khung $Sp(G)$;

6.2 – CÂY BAO TRÙM TỐI TIỂU

Xét lại ví dụ Cho đồ thị vô hướng có trọng số $G=(V, E)$ như bên dưới



$$T = (V, E'),$$

$$V = \{A, B, C, D, E, F\};$$

$$E' = \{(A,E,1), (A,C,1), (C,D,1), (B,D,3), (A,F,4)\} \subseteq E;$$



$$w(T) = 9$$

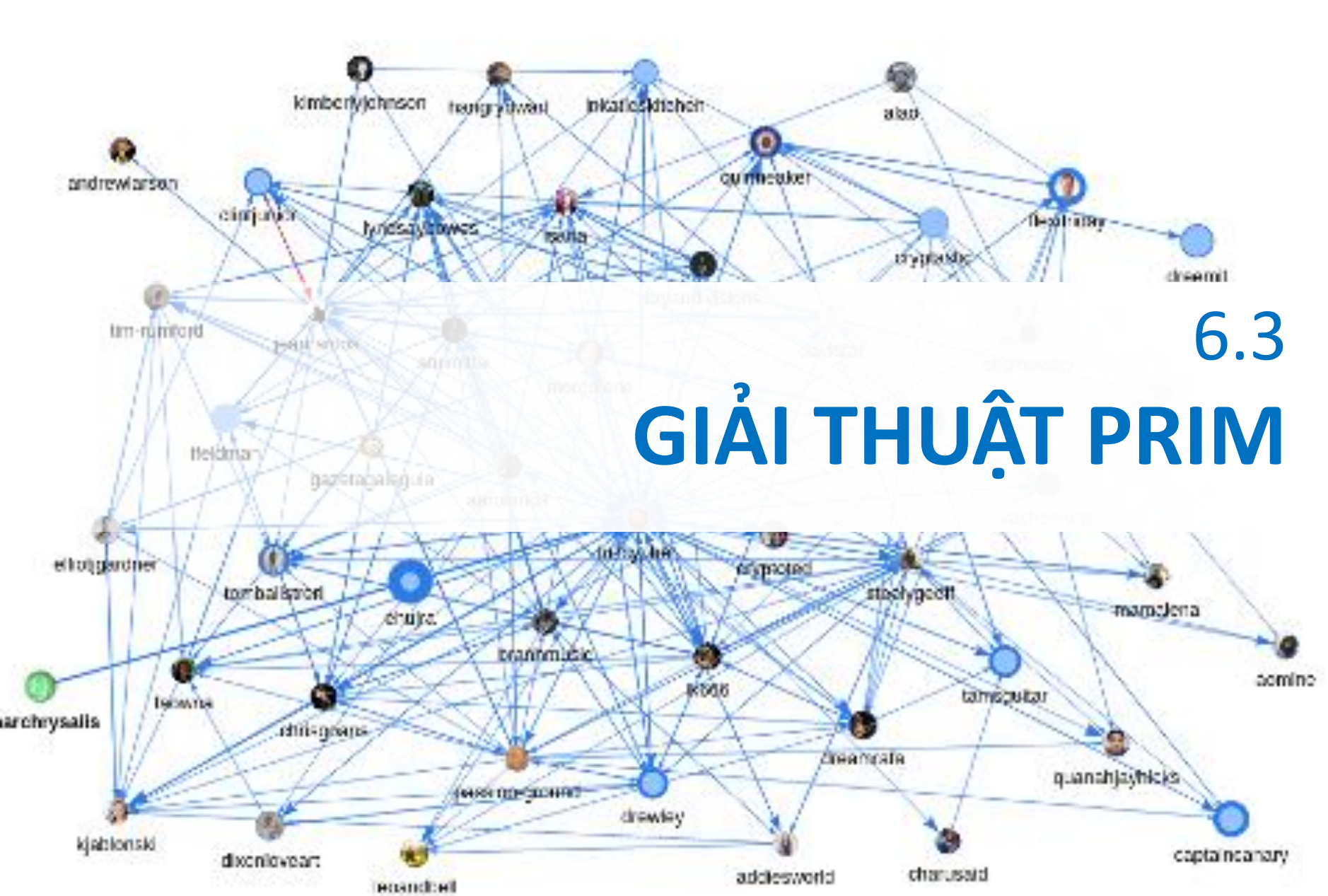
6.2 – CÂY BAO TRÙM TỐI TIỂU

TÌM CÂY BAO TRÙM TỐI TIỂU

6.2 – CÂY BAO TRÙM TỐI TIỂU





BÀI TOÁN TÌM CÂY BAO TRÙM TỐI TIỂU

-  **Đầu vào:** $G = (V, E)$ là một đồ thị liên thông vô hướng có trọng số
-  **Đầu ra:** T (cây bao trùm tối thiểu)



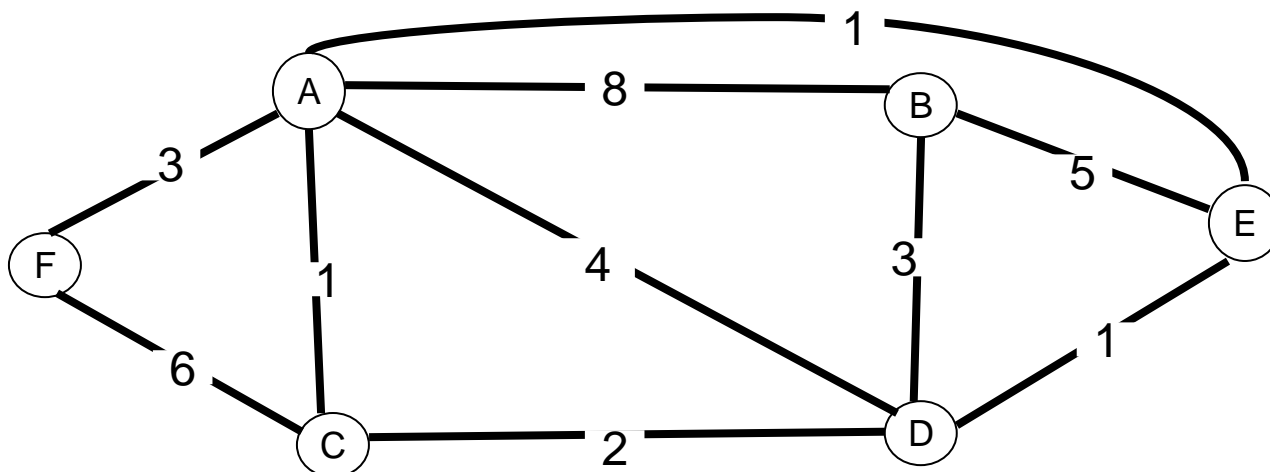
6.3 – THUẬT GIẢI PRIM

Ý TƯỞNG

-  **Bước 0:** bắt đầu từ một đỉnh u bất kì, và gọi u là **đỉnh đang xét**
-  **Bước 1:** tìm tất cả các đỉnh v kề **đỉnh đang xét**, cho các cạnh này vào tập cạnh chuẩn bị xét **Etemp**;
-  **Bước 2:** từ **Etemp** lấy ra một cạnh e , sao cho:
 - $\forall e_i \in \text{Etemp}/\{e\}, w(e) \leq w(e_i)$; ($w(e)$ là trọng số của cạnh e)
 - $\text{Edges}(T) \cup \{e\} \Rightarrow T$ không tạo ra chu trình;
-  **Bước 3:** Nếu không lấy được e nào hoặc **Vertices**(T) = V thì dừng (T là **cây khung tối thiểu**), ngược lại thì gọi $u \in e, u \notin \text{Vertices}(T)$ là **đỉnh đang xét**; quay lại **bước 1**.

6.3 – THUẬT GIẢI PRIM

Minh họa ý tưởng giả sử bắt đầu từ đỉnh A



Đỉnh đang xét



Tập E = ~~{(A,B,8), (A,D,4), (A,E,1), (A,F,3), (B,E,5), (B,D,3), (C,D,2), (C,F,6), (D,E,1), (D,F,4)}~~

Cây T = ~~{(A,C,1), (A,E,1), (E,D,1), (A,F,3)}~~ (D, B, 3) **w(T) = 9**

6.3 – THUẬT GIẢI PRIM

CÀI ĐẶT THUẬT GIẢI PRIM

6.3 – THUẬT GIẢI PRIM

KHAI BÁO CÁC BIẾN TRONG CHƯƠNG TRÌNH

```
// khai bao ma tran bang  
mang hai chieu  
# define MAX 20  
int a[MAX][MAX];  
int n; // so dinh cua do thi
```

```
// khai bao TapE  
int E1[MAX];  
int E2[MAX];  
int wE[MAX];  
int nE=0; // so phan tu tap  
E
```

```
// khai bao TapE  
int T1[MAX];  
int T2[MAX];  
int wT[MAX];  
int nT=0; // so phan tap T
```

6.3 – THUẬT GIẢI PRIM

THỦ TỤC PRIM

```
void prim(int s) // s là đỉnh bắt đầu
{
    int u=s, min, i, d1 d2;
    while(nT<n-1)
    {
        for(int v=0;v<n;v++)
            if(a[u][v]!=0)
                if (TonTai(v, T2, nT)==0)
                {
                    E1[nE]=u; E2[nE]=v;
                    wE[nE]=a[u][v]; nE++;
                }
        for(i=0;i<nE;i++)
            if (TonTai(E2[i], T2, nT)==0)
            {
                min=wE[i]; d1=E1[i];
                d2=E2[i]; break;
            }
    }
```

```
        for(;i<nE;i++)
            if(TonTai(E2[i], T2, nT)==0)
                if(min>wE[i])
                {
                    min=wE[i];
                    d1=E1[i];
                    d2=E2[i];
                }
        T1[nT]=d1; T2[nT]=d2;
        wT[nT]=a[d1][d2];
        a[d1][d2]=0; a[d2][d1]=0;
        nT++;
        XoaCanhE(d1, d2);
        u=d2;
```

}

6.3 – THUẬT GIẢI PRIM

```
int TonTai(int d, int D[], int nD)
{
    for(int i=0;i<nD;i++)
        if(D[i]==d)
            return 1;
    return 0;
}
```


```
void output()
{
    int tong=0;
    for(int i=0;i<nT;i++)
    {
        cout<<endl<<"("<<T1[i]<<","<<T
2[i]<<") = "<<wT[i];
        tong+=wT[i];
    }
    cout<<"\n Tong = "<<tong;
}
```

```
void XoaViTriE(int i)
{
    for(int j=i;j<nE;j++)
    {
        E1[j]=E1[j+1];
        E2[j]=E2[j+1];
        wE[j]=wE[j+1];
    }
    nE--;
}
```

```
void XoaCanhE(int u, int v)
{
    for(int i=0;i<nE;i++)
        if(E1[i]==u&&E2[i]==v)
        {
            XoaViTriE(i);
            break;
        }
}
```

6.3 – THUẬT GIẢI PRIM

ĐỘ PHỨC TẠP THUẬT GIẢI PRIM

 Nếu $G = (V, E)$ là đồ thị liên thông có trọng số; n là số đỉnh của G , và m là số cạnh của G : $n = |V|$, $m = |E|$, thì ta có độ phức tạp của thuật giải Prim là:

$$\approx O(n * \max(n, m))$$



6.4

GIẢI THUẬT KRUSAL

6.4 – THUẬT GIẢI KRUSAL

Ý TƯỞNG THUẬT GIẢI KRUSKAL



Bước 1: từ E lấy ra một cạnh e , sao cho:




- $\forall e_i \in E, w(e) \leq w(e_i)$ ($w(e)$ là trọng số của cạnh e)
- $\text{Edges}(T) \cup \{e\} \Rightarrow T$ không tạo ra chu trình;



Bước 2: Nếu không lấy được e nào hoặc $V = \text{Vertices}(T)$ thì dừng (T là cây khung tối thiểu), ngược lại thì quay lại **bước 1**;

6.4 – THUẬT GIẢI KRUSAL

KRUSKAL CẢI TIẾN

-  **Bước 1:** Sắp xếp E (tăng theo *trọng số* của cạnh)
-  **Bước 2:** Lấy từ E ra một cạnh e , sao cho:
 - $\text{Edges}(T) \cup \{e\} \Rightarrow T$ không tạo ra chu trình;
-  **Bước 3:** $V = \text{Vertices}(T)$ thì dừng (T là *cây khung tối tiểu*), ngược lại thì quay lại **bước 2**;


```

graph LR
    A ---|8| B
    A ---|1| C
    A ---|4| D
    A ---|1| E
    B ---|5| E
    C ---|2| D
    C ---|6| F
    D ---|1| E
    D ---|3| F
    F ---|3| A
  
```

Tập E =

www.ou.edu.vn

6.4 – THUẬT GIẢI KRUSAL

CÀI ĐẶT THUẬT GIẢI KRUSKAL

6.4 – THUẬT GIẢI KRUSAL

KHAI BÁO CÁC BIẾN

```
// khai bao ma tran bang  
mang hai chieu
```

```
# define MAX 20
```

```
int a[MAX][MAX];
```

```
int n; // so dinh cua do thi
```

```
// khai bao TapE
```

```
int E1[MAX];
```

```
int E2[MAX];
```

```
int wE[MAX];
```

```
int nE=0; // so phan tu tap  
E
```

```
// khai bao TapE
```

```
int T1[MAX];
```

```
int T2[MAX];
```

```
int wT[MAX];
```

```
int nT=0; // so phan tap T
```

6.4 – THUẬT GIẢI KRUSAL

THỦ TỤC KRUSKAL

```
void kruskal()  
{  
    for(int i=0;i<nE;i++)  
    {  
        if(TonTai(E1[i], T1, nT)==1 && TonTai(E2[i], T2,  
nT)==1)  
            continue;  
        if (TonTai(E1[i], T2, nT)==1 && TonTai(E2[i], T1,  
nT)==1)  
            continue;  
        T1[nT]=E1[i];  
        T2[nT]=E2[i];  
        wT[nT]=wE[i];  
        nT++;  
        if(nT==n-1)  
            break;  
    }  
}
```


6.4 – THUẬT GIẢI KRUSAL

```
void taoE()  
{  
    for(int i=0;i<n;i++)  
        for(int j=0;j<n;j++)  
            if(a[i][j]!=0)  
            {  
                E1[nE]=i;  
                E2[nE]=j;  
  
                wE[nE]=a[i][j];  
  
                a[i][j]=0;  
                a[j][i]=0;  
  
                nE++;  
            }  
}
```

```
void SapXepE()  
{  
    for(int i=0;i<nE-1;i++)  
        for(int j=i+1;j<nE;j++)  
            if(wE[i]>wE[j])  
            {  
                swap(wE[i],wE[j]);  
  
                swap(E1[i],E1[j]);  
  
                swap(E2[i],E2[j]);  
            }  
}
```

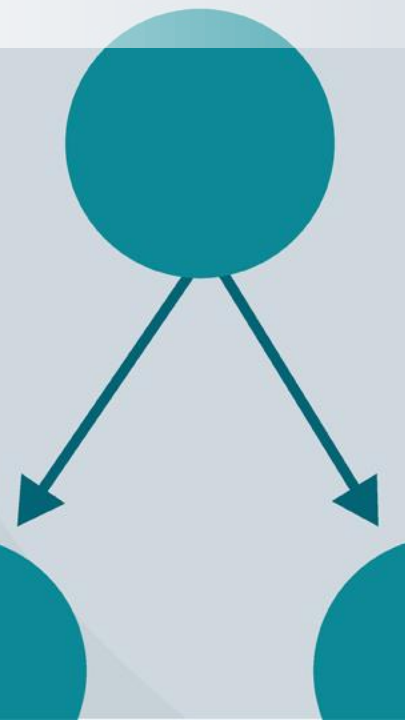
6.4 – THUẬT GIẢI KRUSAL

ĐỘ PHỨC TẠP THUẬT GIẢI

 Nếu $G = (V, E)$ là đồ thị liên thông có trọng số; n là số đỉnh của G , và m là số cạnh của G : $n = |V|$, $m = |E|$, thì ta có độ phức tạp của thuật giải Kruskal là:

$$\approx O(\max(n^2, m^2))$$

6.5 – TỔNG KẾT CHƯƠNG 6



5.5 – Tổng kết chương 5



Cây bao trùm



Cây bao trùm tối thiểu



Giải thuật tìm kiếm PRIM



Giải thuật tìm kiếm KRUSAL

A person wearing a blue and white checkered shirt is sitting at a wooden desk, working on a silver laptop. The laptop screen displays a website with a car image. To the left of the laptop is an open white notebook with a pen resting on it. To the right is a light blue mug filled with coffee. In the background, there is a stack of books and a black container. The background wall is made of white bricks.

6.6- Bài tập rèn luyện **CHƯƠNG 6**

6.6 - Bài tập chương 6

CÂU HỎI



Câu 1: Cây bao trùm là gì? Cho ví dụ? Cây bao trùm tối thiểu là gì?



Câu 2: Đồ thị và cây bao trùm có những điểm giống và khác nhau như thế nào?

6.6 - Bài tập chương 6

BÀI TẬP LÝ THUYẾT

Bài 1: Trình bày ý tưởng của thuật giải Prim

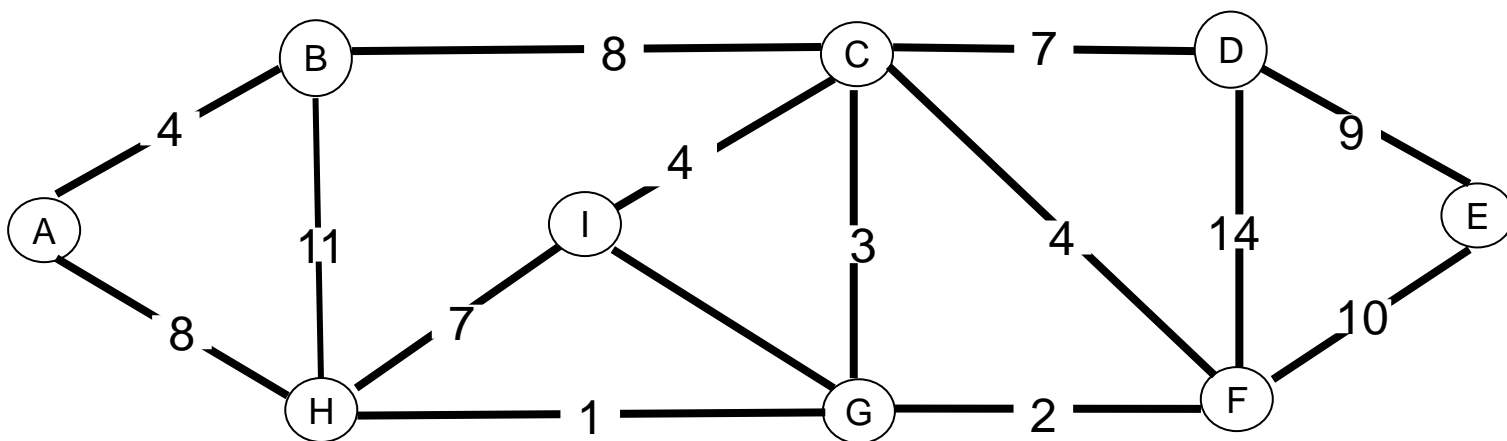
Bài 2: Trình bày ý tưởng của thuật giải Kruskal

Bài 3: Hãy cho biết sự khác biệt giữa hai ý tưởng của thuật giải Prim và thuật giải Kruskal

6.6 - Bài tập chương 6

BÀI TẬP THỰC HÀNH

Đồ thị $G = (V, E)$



6.6 - Bài tập chương 6

BÀI TẬP THỰC HÀNH

Bài 4: Viết chương trình thực hiện tìm cây khung tối thiểu cho đồ thị G bằng thuật giải Prim. (*dựa trên phương pháp biểu diễn đồ thị bằng ma trận kề*)

Bài 5: Viết chương trình thực hiện tìm cây khung tối thiểu của G bằng thuật giải Kruskal. (*dựa trên phương pháp biểu diễn đồ thị bằng ma trận kề*)

Bài 6: Viết chương trình thực hiện tìm cây khung tối thiểu của G bằng thuật giải Kruskal cải tiến. (*dựa trên phương pháp biểu diễn đồ thị bằng ma trận kề*)

6.6 - Bài tập chương 6

BÀI TẬP LÀM THÊM

Bài 7: Thực hiện biểu diễn G (bằng ma trận kề) lên trên máy tính và đặt tên là do_thi_1.txt (dạng file TEXT) sau đó thực hiện viết chương trình tìm cây khung tối thiểu cho đồ thị G:

7.1. Bảng thuật giải Prim

7.2. Bảng thuật giải Kruskal

7.2. Bảng thuật giải Kruskal cải tiến

Hướng dẫn

- Tất cả sinh viên phải trả lời các **câu hỏi** của chương, làm **bài tập thực hành** tại phòng máy (**bài làm thêm** ở nhà, và **bài nâng cao** khuyến khích hoàn tất) và nộp bài qua LMS của trường.
- Câu hỏi và bài tập lý thuyết chương 6 làm trên file WORD; trong bài làm ghi rõ họ tên, lớp, bài tập chương và các thông tin cần thiết.
- Khuyến khích sử dụng tiếng Anh trong bài tập.

⇒ **Ngày nộp:** trước khi kết thúc môn học.

⇒ **Cách nộp:** sử dụng **github** để nộp bài, sau đó nộp lên LMS của trường.

Tài liệu tham khảo

- **Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest, Clifford Stein**, (Chapter 23) *Introduction to Algorithms*, Third Edition, 2009.
- **Judith I. Gersting**, (Chapter 6&7) *mathematical structures for computer science*, 2014.

Phụ lục – Thuật ngữ tiếng Anh

#	Tiếng Anh	Phiên Âm	Tiếng Việt
1	Spanning Tree	/ 'spæniŋ tri: /	Cây bao trùm / Cây khung
2	Minimum Spanning Tree	/ 'miniməm 'spæniŋ tri:/	Cây bao trùm tối thiểu

KẾT THÚC CHƯƠNG 6



Trường Đại học Mở TP.HCM

Khoa Công Nghệ Thông Tin