

# Implémentation d'un ABR

# le header correspondant à l'ABR

AB PlusPetit();

# le header correspondant à l'ABR

```
AB PlusPetit();  
AB RechercherValeur(Valeur v);
```

# le header correspondant à l'ABR

```
AB PlusPetit();  
AB RechercherValeur(Valeur v);  
void InsérerValeur(Valeur v);
```

# le header correspondant à l'ABR

```
AB PlusPetit();  
AB RechercherValeur(Valeur v);  
void InsérerValeur(Valeur v);
```

```
AB SupMin();
```

# le header correspondant à l'ABR

```
AB PlusPetit();  
AB RechercherValeur(Valeur v);  
void InsérerValeur(Valeur v);
```

```
AB SupMin();  
AB SupprimerValeur(Valeur v);
```

# le header correspondant à l'ABR

```
AB PlusPetit();  
AB RechercherValeur(Valeur v);  
void InsérerValeur(Valeur v);
```

```
AB SupMin();  
AB SupprimerValeur(Valeur v);  
    // notez la dissymétrie
```

# le header correspondant à l'ABR

```
AB PlusPetit();  
AB RechercherValeur(Valeur v);  
void InsérerValeur(Valeur v);  
    // on insère toujours une nouvelle feuille dans une  
    arborescence non vide  
AB SupMin();  
AB SupprimerValeur(Valeur v);  
    // notez la dissymétrie  
// l'adresse de la racine de l'arborescence est éventuellement  
modifiée
```



```
AB Sommet : :PlusPetit(){  
  
}
```

```
AB Sommet : :PlusPetit(){  
    if (SAG)  
        }
```

```
AB Sommet : :PlusPetit(){  
    if (SAG) return SAG->PlusPetit();  
}
```

```
AB Sommet : :PlusPetit(){  
    if (SAG) return SAG->PlusPetit();  
    return this; }
```

# les algorithmes simples de manipulation de l'ABR : RechercherValeur

```
AB Sommet : :RechercherValeur(Valeur v){  
    if (cle==v) return this;  
  
    }  
}
```

# les algorithmes simples de manipulation de l'ABR : RechercherValeur

```
AB Sommet : :RechercherValeur(Valeur v){  
    if (cle==v) return this;  
    if (cle>v)  
        {  
                                                    }  
    else  
        {  
                                                    }  
}
```

# les algorithmes simples de manipulation de l'ABR : RechercherValeur

```
AB Sommet : :RechercherValeur(Valeur v){  
    if (cle==v) return this;  
    if (cle>v)  
        { if (SAG) return  
          else return      }  
    else  
        {  
        }  
}
```

# les algorithmes simples de manipulation de l'ABR : RechercherValeur

```
AB Sommet : :RechercherValeur(Valeur v){  
    if (cle==v) return this;  
    if (cle>v)  
        { if (SAG) return SAG→RechercherValeur(v);  
          else return      }  
    else  
        {  
            }  
}
```



# les algorithmes simples de manipulation de l'ABR : RechercherValeur

```
AB Sommet : :RechercherValeur(Valeur v){  
    if (cle==v) return this;  
    if (cle>v)  
        { if (SAG) return SAG→RechercherValeur(v);  
          else return NULL }  
    else  
        {  
        }  
}
```

# les algorithmes simples de manipulation de l'ABR : RechercherValeur

```
AB Sommet : :RechercherValeur(Valeur v){  
    if (cle==v) return this;  
    if (cle>v)  
        { if (SAG) return SAG→RechercherValeur(v);  
          else return NULL }  
    else  
        { if (SAD) return  
          else return      }  
}
```

# les algorithmes simples de manipulation de l'ABR : RechercherValeur

```
AB Sommet : :RechercherValeur(Valeur v){  
    if (cle==v) return this;  
    if (cle>v)  
        { if (SAG) return SAG→RechercherValeur(v);  
          else return NULL }  
    else  
        { if (SAD) return SAD→RechercherValeur(v);  
          else return      }  
}
```

# les algorithmes simples de manipulation de l'ABR : RechercherValeur

```
AB Sommet : :RechercherValeur(Valeur v){  
    if (cle==v) return this;  
    if (cle>v)  
        { if (SAG) return SAG→RechercherValeur(v);  
          else return NULL }  
    else  
        { if (SAD) return SAD→RechercherValeur(v);  
          else return NULL; }  
}
```

## les algorithmes simples de manipulation de l'ABR : InsérerValeur

```
void Sommet : :InsererValeur(Valeur v){
```

# les algorithmes simples de manipulation de l'ABR : InsérerValeur

```
void Sommet : :InsérerValeur(Valeur v){  
    if (cle  $\geq$  v)  
        {  
                                                    }  
    else  
        {  
                                                    }  
}
```

# les algorithmes simples de manipulation de l'ABR : InsérerValeur

```
void Sommet : :InsérerValeur(Valeur v){  
    if (cle  $\geq$  v)  
        { if (SAG)  
            else  
        }  
    else  
        {  
        }  
}
```

# les algorithmes simples de manipulation de l'ABR : InsérerValeur

```
void Sommet : :InsérerValeur(Valeur v){  
    if (cle  $\geq$  v)  
        { if (SAG) SAG  $\rightarrow$  InsérerValeur(v);  
          else  
        }  
    else  
        {  
        }  
}
```



# les algorithmes simples de manipulation de l'ABR : InsérerValeur

```
void Sommet : :InsérerValeur(Valeur v){  
    if (cle  $\geq$  v)  
        { if (SAG) SAG  $\rightarrow$  InsérerValeur(v);  
          else GrefferSAG(new Sommet(v)); }  
    else  
        {  
                                                    }  
}
```

# les algorithmes simples de manipulation de l'ABR : InsérerValeur

```
void Sommet : :InsérerValeur(Valeur v){  
    if (cle  $\geq$  v)  
        { if (SAG) SAG  $\rightarrow$  InsérerValeur(v);  
          else GrefferSAG(new Sommet(v)); }  
    else  
        { if (SAD)  
          else  
        }  
}
```

# les algorithmes simples de manipulation de l'ABR : InsérerValeur

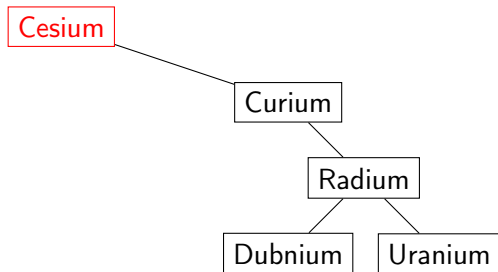
```
void Sommet : :InsérerValeur(Valeur v){  
    if (cle ≥ v)  
        { if (SAG) SAG → InsérerValeur(v);  
          else GrefferSAG(new Sommet(v)); }  
    else  
        { if (SAD) SAD → InsérerValeur(v);  
          else  
        }  
}
```

# les algorithmes simples de manipulation de l'ABR : InsérerValeur

```
void Sommet : :InsérerValeur(Valeur v){  
    if (cle $\geq$ v)  
        { if (SAG) SAG $\rightarrow$ InsérerValeur(v);  
          else GrefferSAG(new Sommet(v)); }  
    else  
        { if (SAD) SAD $\rightarrow$ InsérerValeur(v);  
          else GrefferSAD(new Sommet(v)); }  
}
```

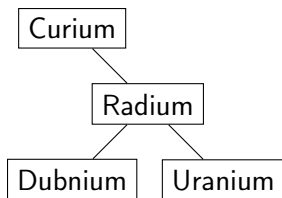
# les algorithmes simples de manipulation de l'ABR : SupMin

le cas le plus simple



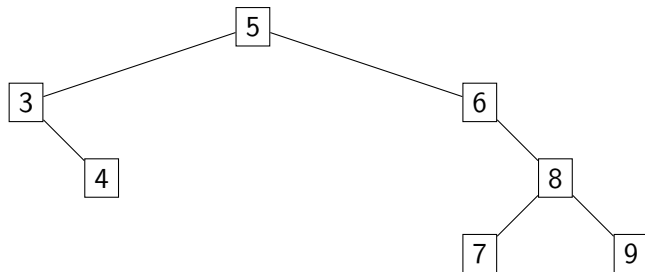
# les algorithmes simples de manipulation de l'ABR : SupMin

le cas le plus simple



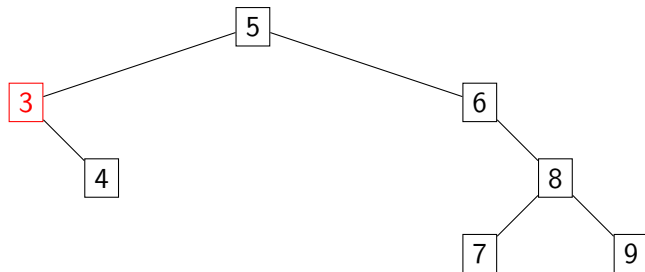
# les algorithmes simples de manipulation de l'ABR : SupMin

le cas à peine plus compliqué



# les algorithmes simples de manipulation de l'ABR : SupMin

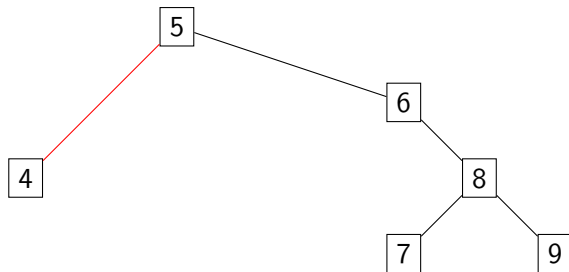
le cas à peine plus compliqué





# les algorithmes simples de manipulation de l'ABR : SupMin

le cas à peine plus compliqué



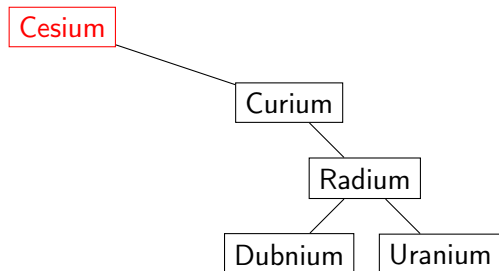
# les algorithmes simples de manipulation de l'ABR : SupMin l'algo

AB Sommet : :SupMin(){

}

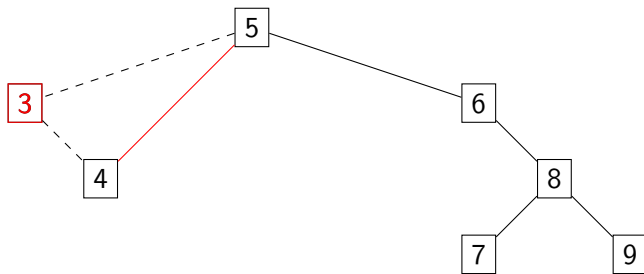
# les algorithmes simples de manipulation de l'ABR : SupMin l'algo

```
AB Sommet : :SupMin(){  
    if ( !SAG) {  
        return SAD}
```



# les algorithmes simples de manipulation de l'ABR : SupMin l'algo

```
AB Sommet : :SupMin(){  
    if ( !SAG) {if (Pere) RemplacerPourLePerePar(SAD);  
        return SAD}
```



# les algorithmes simples de manipulation de l'ABR : SupMin l'algo

```
AB Sommet : :SupMin(){  
    if ( !SAG) {if (Pere) RemplacerPourLePerePar(SAD);  
        return SAD}  
    AB min=PlusPetit();
```

```
}
```

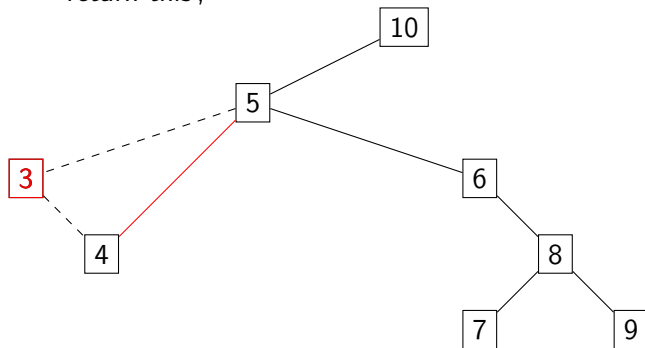
# les algorithmes simples de manipulation de l'ABR : SupMin l'algo

```
AB Sommet : :SupMin(){  
    if ( !SAG) {if (Pere) RemplacerPourLePerePar(SAD);  
        return SAD}  
    AB min=PlusPetit();  
    min→RemplacerPourLePerePar(min->SAD);  
    return this;  
}
```

}

# les algorithmes simples de manipulation de l'ABR : SupMin l'algo

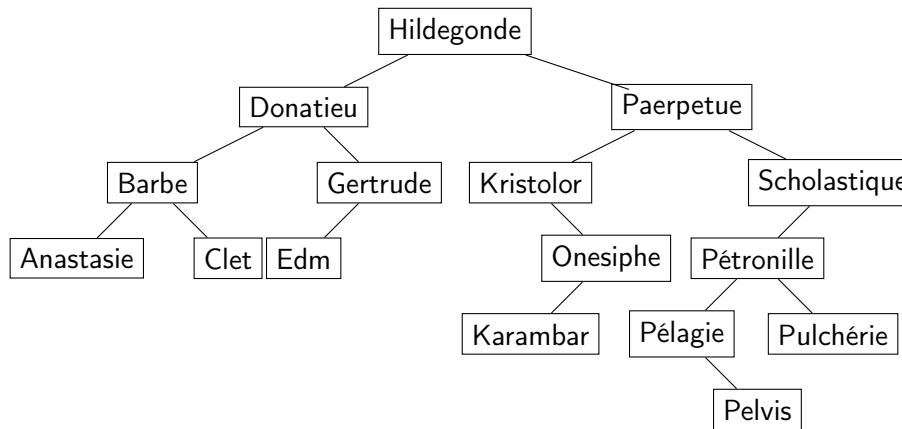
```
AB Sommet : :SupMin(){  
    if ( !SAG) {if (Pere) RemplacerPourLePerePar(SAD);  
        return SAD}  
    AB min=PlusPetit();  
    min→RemplacerPourLePerePar(min->SAD);  
    return this;  
}
```



}

# suppression d'un élément de l'ABR

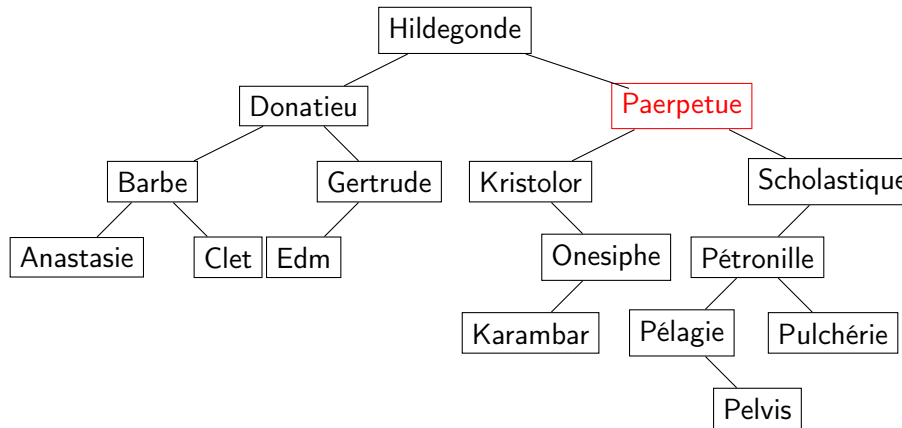
un exemple





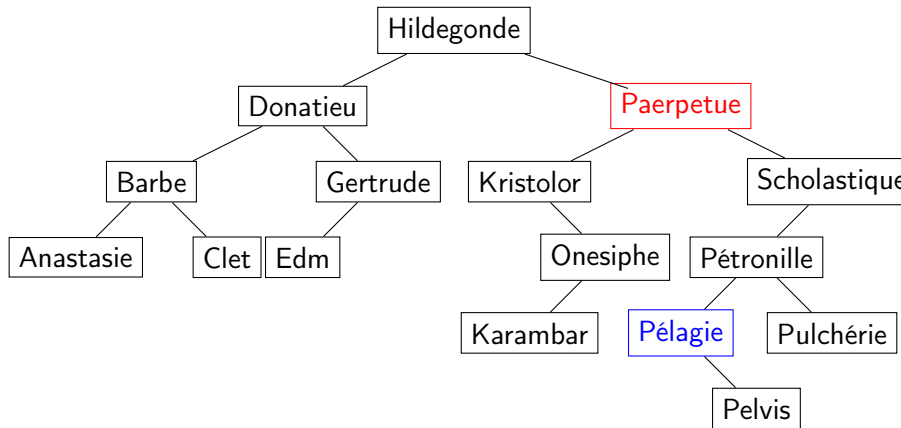
# suppression d'un élément de l'ABR

un exemple



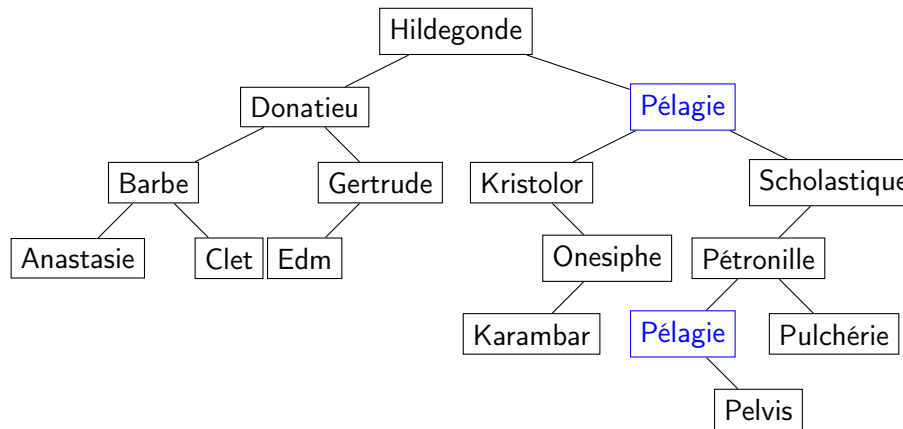
# suppression d'un élément de l'ABR

un exemple



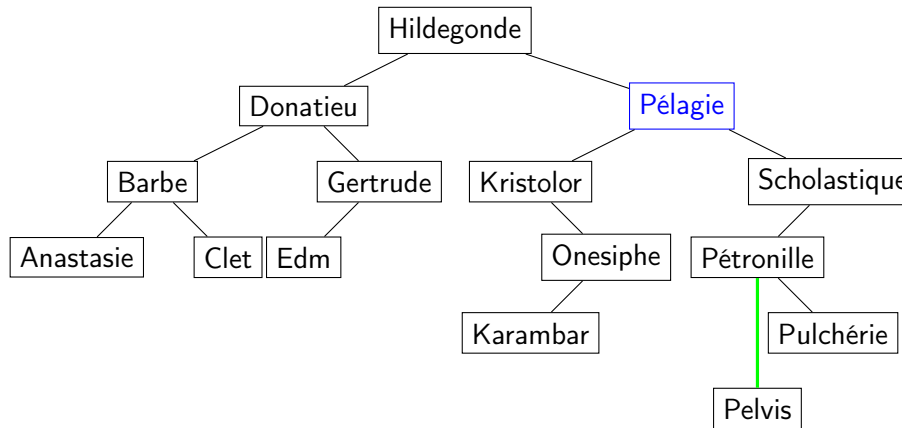
# suppression d'un élément de l'ABR

un exemple



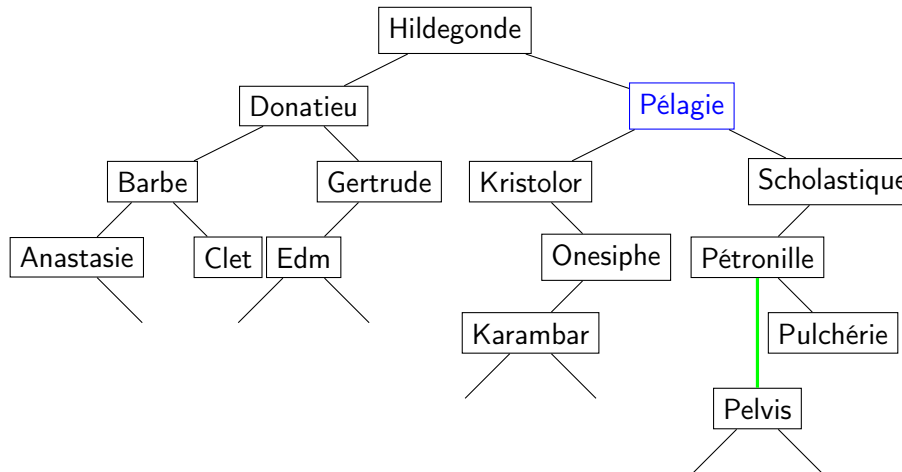
# suppression d'un élément de l'ABR

un exemple



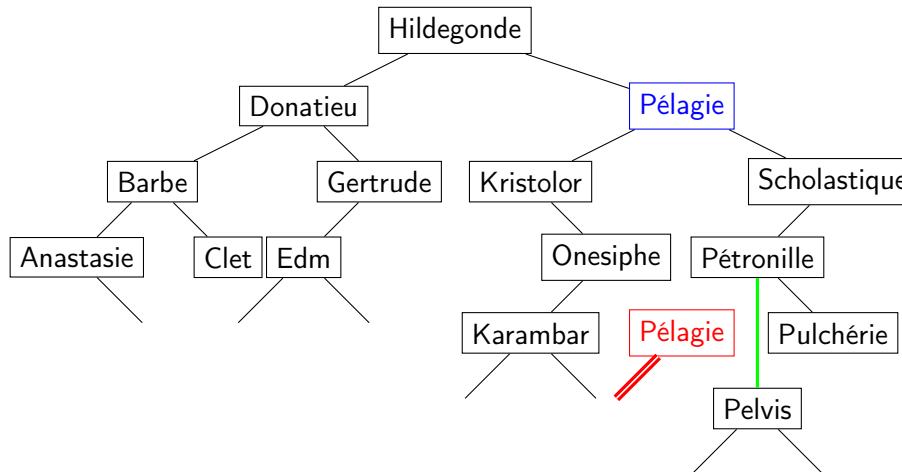
# suppression d'un élément de l'ABR

un exemple



# suppression d'un élément de l'ABR

un exemple



# suppression d'un élément de l'ABR

## l'algorithme

AB Sommet : :SupprimerValeur(Valeur v){

}

# suppression d'un élément de l'ABR

## l'algorithme

```
AB Sommet : :SupprimerValeur(Valeur v){  
    AB r=RechercherValeur(v) ;
```

```
}
```



# suppression d'un élément de l'ABR

## l'algorithme

```
AB Sommet : :SupprimerValeur(Valeur v){  
    AB r=RechercherValeur(v);  
    if (!r) return this;  
  
}
```

# suppression d'un élément de l'ABR

## l'algorithme

```
AB Sommet : :SupprimerValeur(Valeur v){  
    AB r=RechercherValeur(v);  
    if ( !r) return this;  
    if (r→FeuilleP()) {  
        }  
    }
```

```
}
```

# suppression d'un élément de l'ABR

## l'algorithme

```
AB Sommet : :SupprimerValeur(Valeur v){  
    AB r=RechercherValeur(v);  
    if (!r) return this;  
    if (r→FeuilleP()) { if (!r→Pere) return NULL;  
                        }  
}
```

# suppression d'un élément de l'ABR

## l'algorithme

```
AB Sommet : :SupprimerValeur(Valeur v){  
    AB r=RechercherValeur(v);  
    if (!r) return this;  
    if (r→FeuilleP()) { if (!r→Pere) return NULL;  
        r→RemplacerPourLePerePar(NULL);  
    }  
  
}
```

# suppression d'un élément de l'ABR

## l'algorithme

```
AB Sommet : :SupprimerValeur(Valeur v){  
    AB r=RechercherValeur(v);  
    if (!r) return this;  
    if (r→FeuilleP()) { if (!r→Pere) return NULL;  
        r→RemplacerPourLePerePar(NULL); return this; }  
  
}
```

## suppression d'un élément de l'ABR

```

AB Sommet : :SupprimerValeur(Valeur v){
    AB r=RechercherValeur(v);
    if (!r) return this;
    if (r→FeuilleP()) { if (!r→Pere) return NULL;
        r→RemplacerPourLePerePar(NULL); return this; }
    if (!r→SAG) {
        }
    if (!r→SAD) {
        }
}

//et d'autre part le cas général
}

```

# suppression d'un élément de l'ABR

## l'algorithme

```
AB Sommet : :SupprimerValeur(Valeur v){
    AB r=RechercherValeur(v);
    if (!r) return this;
    if (r→FeuilleP()) { if (!r→Pere) return NULL;
        r→RemplacerPourLePerePar(NULL); return this; }
    if (!r→SAG) { if (r→Pere)
    {
        else {
        if (!r→SAD) {
        }
    }
    }
    //et d'autre part le cas général
}
```

# suppression d'un élément de l'ABR

## l'algorithme

```
AB Sommet : :SupprimerValeur(Valeur v){
    AB r=RechercherValeur(v);
    if (!r) return this;
    if (r→FeuilleP()) { if (!r→Pere) return NULL;
        r→RemplacerPourLePerePar(NULL); return this; }
    if (!r→SAG) { if (r->Pere)
    {
        else {r->SAD->Pere=NULL; return r->SAD;} }
    if (!r→SAD) {

    }

    //et d'autre part le cas général

}
```



# suppression d'un élément de l'ABR

## l'algorithme

```
AB Sommet : :SupprimerValeur(Valeur v){
    AB r=RechercherValeur(v);
    if (!r) return this;
    if (r→FeuilleP()) { if (!r→Pere) return NULL;
        r→RemplacerPourLePerePar(NULL); return this; }
    if (!r→SAG) { if (r->Pere)
        {r->RemplacerPourLePerePar(r->SAD); return this;}
        else {r->SAD->Pere=NULL; return r->SAD;} }
    if (!r→SAD) {

    }
```

//et d'autre part le cas général

```
}
```

# suppression d'un élément de l'ABR

## l'algorithme

```
AB Sommet : :SupprimerValeur(Valeur v){
    AB r=RechercherValeur(v);
    if (!r) return this;
    if (r→FeuilleP()) { if (!r→Pere) return NULL;
        r→RemplacerPourLePerePar(NULL); return this; }
    if (!r→SAG) { if (r->Pere)
        {r->RemplacerPourLePerePar(r->SAD); return this;}
        else {r->SAD->Pere=NULL; return r->SAD;} }
    if (!r→SAD) { if (r->Pere)
        {r->RemplacerPourLePerePar(r->SAG); return this;}
        else {r->SAG->Pere=NULL; return r->SAG;} }
    //et d'autre part le cas général

}
```

# suppression d'un élément de l'ABR

## l'algorithme

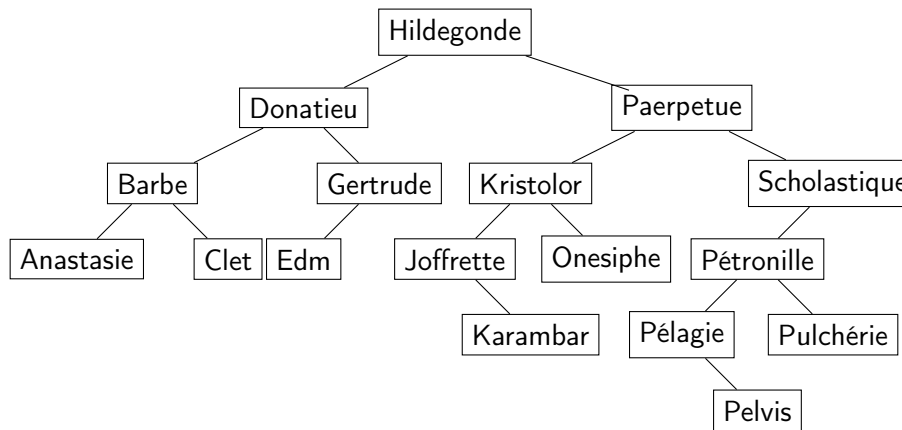
```
AB Sommet : :SupprimerValeur(Valeur v){
    AB r=RechercherValeur(v);
    if (!r) return this;
    if (r→FeuilleP()) { if (!r→Pere) return NULL;
        r→RemplacerPourLePerePar(NULL); return this; }
    if (!r→SAG) {
        }

    if (!r→SAD) {
        }

    //et d'autre part le cas général
    r→valeur= r→SAD→PlusPetit()→valeur;
    r→SAD→SupMin(); return this;
}
```

# suppression d'un élément de l'ABR

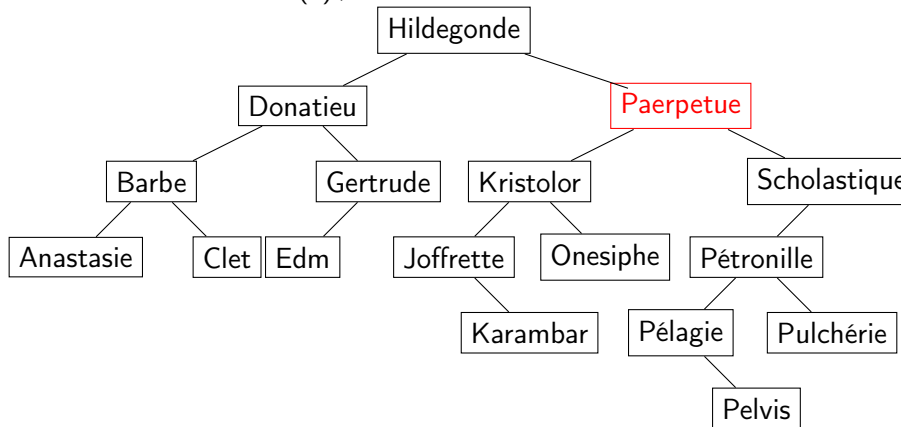
trace de l'algorithme sur l'exemple



# suppression d'un élément de l'ABR

trace de l'algorithme sur l'exemple

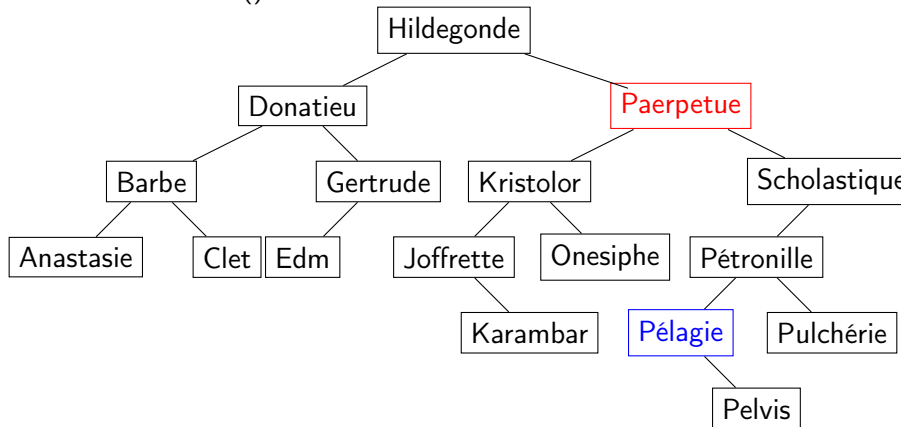
AB r=RechercherValeur(v) ;



# suppression d'un élément de l'ABR

trace de l'algorithme sur l'exemple

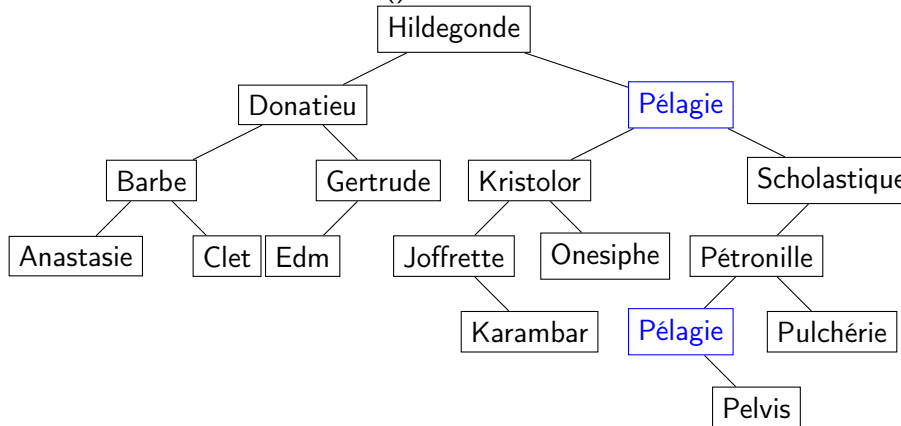
$r \rightarrow \text{SAD} \rightarrow \text{PlusPetit}()$



# suppression d'un élément de l'ABR

trace de l'algorithme sur l'exemple

$r \rightarrow \text{valeur} = r \rightarrow \text{SAD} \rightarrow \text{PlusPetit}() \rightarrow \text{valeur};$



# suppression d'un élément de l'ABR

trace de l'algorithme sur l'exemple

$r \rightarrow \text{SAD} \rightarrow \text{SupMin}(); \text{return this};$

