



### PROGRAMMATION DE MACHINES À L'ESI

Mikael Degeer

École Supérieure d'Informatique



```

graph TD
    subgraph "graph"
        direction TB
        G[Graph]
        O[Observer]
        L[Link]
        N[Node]
        T[Thread]
        M[Matrix]
        V[Vector]
        F[Function]
    end
    class G public TThread
    class O public TObserver
    class L public TLink
    class N public TNode
    class M public TMatrix
    class V public TVector
    class F public TFunction
  
```

12

10

```

void _fastcall TNode::TNode(const TNode* n)
{
    this->SetParent(n);
    this->SetColor(n->Color);
    this->SetX(n->X);
    this->SetY(n->Y);
    this->SetLabel(n->Label);
    this->SetType(n->Type);
    this->SetWeight(n->Weight);
    this->SetVisible(n->Visible);
}

void TNode::SetParent(const TNode* n)
{
    this->Parent = n;
}

void TNode::SetColor(const TColor* c)
{
    this->Color = *c;
}

void TNode::SetX(const TPoint* p)
{
    this->X = p->X;
}

void TNode::SetY(const TPoint* p)
{
    this->Y = p->Y;
}

void TNode::SetLabel(const TText* t)
{
    this->Label = *t;
}

void TNode::SetType(const TType* t)
{
    this->Type = *t;
}

void TNode::SetWeight(const TWeight* w)
{
    this->Weight = *w;
}

void TNode::SetVisible(const TBool* b)
{
    this->Visible = *b;
}
  
```

8

7

```

Afficher le graphe? (A)
Renome le graphe? (B)
Ajouter un Noeud? (C)
Supprimer un Noeud? (D)
Afficher le nombre de Noeuds? (E)
Ajouter un Arc? (F)
Supprimer un Arc? (G)
Afficher le nombre d'Arcs? (H)
Matrice d'accessibilite? (I)
Matrice de Marshall? (J)
Matrice d'accessibilite? (K)
Vecteur de TNoeud? (L)
Vecteur de TArce? (M)
Le plus court chemin? (N)
Quitter? (Q)
  
```

5

4

```

class Graf
{
private:
    class TArce;
    class TNoeud;
public:
    TNoeud(string nom, TNoeud* svt=NULL, TArce* lien=NULL)
    {
        Nom=nom;svt=svt;lien=lien;
        string Nom;
        TNoeud* sv;
        TArce* Lien;
    };

    class TArce{
public:
    TArce(TNoeud* destination=NULL, TArce* suivant=NULL)
    {
        Destination=destination; Suivant=suivant;
        TArce* Suivant;
    };

public:
    Graf(string temp="Graphe 1");
    string NomGraphe();
    void ChangeNomGraphe(string);

    int AjouteNoeud(string);
    bool NoeudExiste(string);
    int SupprimeNoeud(string);
    int NombreDeNoeuds();
  
```

1

2

Toutes les photographies représentées ici proviennent de Mikael Degeer.