



# Programmation de machines à l'ESI

École Supérieure d'Informatique (Haute École de Bruxelles)

2<sup>es</sup> années

HAUTE ÉCOLE DE BRUXELLES – ÉCOLE SUPÉRIEURE D'INFORMATIQUE

```

class WinLabelObserver : public Observateur
{
public:
    WinLabelObserver() {}
    virtual void observer(TComponent * proprio);
    virtual void _fastcall WinObs(TComponent * proprio);
};

class WinObs : public TForm
{
public:
    TLabel * info;
    virtual _fastcall WinObs(TComponent * proprio);
    virtual void _fastcall WinObs(TComponent * proprio);
};

class WinLabelObserver : public Observateur
{
public:
    WinLabelObserver() {}
    virtual void observer(TComponent * proprio);
    virtual void _fastcall WinObs(TComponent * proprio);
};
    
```

```

void _fastcall WinObs::SupprimerNoeud(TComponent * proprio)
{
    this->info->Text="";
    this->info->Color=clNone;
    this->info->Visible=false;
    this->info->Parent=this;
    this->info->Free();
}

void WinObs::SupprimerNoeud()
{
    this->info->Text="";
    this->info->Color=clNone;
    this->info->Visible=false;
    this->info->Parent=this;
    this->info->Free();
}
    
```

- 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 TArCs? (M)
- Le plus court chemin? (N)
- Quitter? (O)

```

class Graf
{
private:
    class TArCi;
    class TNoeudi;
public:
    TNoeudi(string nom, TNoeudi* svt=NULL, TArCi* lien=NULL)
    {
        Nom=nom;Svt=svt;Lien=lien;
    }
    string Nom;
    TNoeudi* Svt;
    TArCi* Lien;
};

class TArCi
{
    friend class Graf::TNoeudi;
public:
    TArCi(TNoeudi* destination=NULL,TArCi* suivant=NULL)
    {
        Destination=destination; Suivant=suivant;
    }
    TNoeudi* Destination;
    TArCi* Suivant;
};

public:
    Graf(string temp="Graphe 1");
    string NomGraphe() {return NomGraf;}
    void ChangerNomGraphe(string);

    int AjouterNoeud(string);
    bool NoeudExiste(string);
    int SupprimerNoeud(string);
    int NombreDeNoeuds();
};
    
```

Toutes les photographies représentées ici proviennent de Mikael Degeer.