

```

#include <ilcplex/ilocplex.h>

ILOSTLBEGIN;

//LES MATRICES PRE-REMPLES DANS CE CODE CORRESPONDENT A CELLES UTILISE POUR AVOIR 77.56%
//(voir image capture.png)

int main()
{
    //Creation de l'environnement
    IloEnv env;
    try
    {
        ////////////////////////////////////////VARIABLE//////////////////////////////////////

        //creation du model du probleme
        IloModel model(env);

        //definition des different types de tableaux utiles pour resoudre le probleme
        typedef IloArray<IloNumArray> IloNumArray2;
        typedef IloArray<IloNumArray2> IloNumArray3;
        typedef IloArray<IloNumVarArray> IloNumVarArray2;
        typedef IloArray<IloNumVarArray2> IloNumVarArray3;
        typedef IloArray<IloNumVarArray3> IloNumVarArray4;

        //definition des variables utiles
        IloInt nb_chauffeurs=11;
        IloInt nb_lignes=3;
        IloInt nb_jours=14;

        //Remplissage des lignes de bus pour chaque conducteur
        //0->ne s'occupe pas de cette ligne
        //1->s'occupe de cette ligne
        //la premiere case correspond a la ligne 1, la seconde a la ligne 2 et la troisieme
        a la ligne 3
        //REEMPLIR EN DUR CETTE MATRICE
        IloNumArray2 affectation_lignes(env,nb_chauffeurs);
        affectation_lignes[0]=IloNumArray(env,nb_lignes,1,0,0);
        affectation_lignes[1]=IloNumArray(env,nb_lignes,0,1,0);
        affectation_lignes[2]=IloNumArray(env,nb_lignes,0,1,1);
        affectation_lignes[3]=IloNumArray(env,nb_lignes,0,1,0);
        affectation_lignes[4]=IloNumArray(env,nb_lignes,1,0,0);
        affectation_lignes[5]=IloNumArray(env,nb_lignes,0,0,1);
        affectation_lignes[6]=IloNumArray(env,nb_lignes,1,1,0);
        affectation_lignes[7]=IloNumArray(env,nb_lignes,1,0,1);
        affectation_lignes[8]=IloNumArray(env,nb_lignes,1,0,0);
        affectation_lignes[9]=IloNumArray(env,nb_lignes,0,1,0);
        affectation_lignes[10]=IloNumArray(env,nb_lignes,0,1,1);

        //Remplissage des cases noires(0 pour une case noire, 1 pour une autre case)
        //REEMPLIR EN DUR CETTE MATRICE
        IloNumArray2 disponibilite_conducteur(env,nb_chauffeurs);
        disponibilite_conducteur[0]=IloNumArray(env,nb_jours,0,1,1,1,1,1,0,0,1,1,1,1,1,0);
        disponibilite_conducteur[1]=IloNumArray(env,nb_jours,0,0,1,1,1,1,1,0,0,1,1,1,1,1);
        disponibilite_conducteur[2]=IloNumArray(env,nb_jours,1,0,0,1,1,1,1,1,0,0,1,1,1,1);
        disponibilite_conducteur[3]=IloNumArray(env,nb_jours,1,1,0,0,1,1,1,1,1,0,0,1,1,1);
        disponibilite_conducteur[4]=IloNumArray(env,nb_jours,1,1,1,0,0,1,1,1,1,1,0,0,1,1);
        disponibilite_conducteur[5]=IloNumArray(env,nb_jours,1,1,1,1,0,0,1,1,1,1,1,0,0,1);
        disponibilite_conducteur[6]=IloNumArray(env,nb_jours,1,1,1,1,1,0,0,1,1,1,1,1,0,0);
        disponibilite_conducteur[7]=IloNumArray(env,nb_jours,0,1,1,1,1,1,0,0,1,1,1,1,1,0);
        disponibilite_conducteur[8]=IloNumArray(env,nb_jours,0,0,1,1,1,1,1,0,0,1,1,1,1,1);
        disponibilite_conducteur[9]=IloNumArray(env,nb_jours,1,0,0,1,1,1,1,1,0,0,1,1,1,1);
        disponibilite_conducteur[10]=IloNumArray(env,nb_jours,1,1,0,0,1,1,1,1,1,0,0,1,1,1);

        //Remplissage des cases grises(1 pour une case grise, 0 pour une autre case)
        //REEMPLIR EN DUR CETTE MATRICE
        IloNumArray2 conges_demandes(env, nb_chauffeurs);
        conges_demandes[0]=IloNumArray(env,nb_jours,0,0,0,1,0,0,0,0,0,1,0,1,0,0);
        conges_demandes[1]=IloNumArray(env,nb_jours,0,0,0,0,0,1,0,0,0,0,1,0,1,0);
        conges_demandes[2]=IloNumArray(env,nb_jours,0,0,0,0,1,0,1,0,0,0,0,0,0,0);
        conges_demandes[3]=IloNumArray(env,nb_jours,0,0,0,0,0,0,0,0,0,0,0,0,1,0);
    }
}

```

```

conges_demandes[4]=IloNumArray(env,nb_jours,0,1,0,0,0,0,1,0,0,0,0,0,0,1);
conges_demandes[5]=IloNumArray(env,nb_jours,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0);
conges_demandes[6]=IloNumArray(env,nb_jours,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0);
conges_demandes[7]=IloNumArray(env,nb_jours,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0);
conges_demandes[8]=IloNumArray(env,nb_jours,0,0,0,0,1,0,0,0,0,0,0,1,0,1,0);
conges_demandes[9]=IloNumArray(env,nb_jours,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0);
conges_demandes[10]=IloNumArray(env,nb_jours,0,0,0,0,0,0,1,1,0,0,0,0,1,0,0);

```

```
//Creation du tableau 3 dimension pour les cases jaunes
```

```
IloNumArray3 travaux_demandes(env,nb_chauffeurs);
```

```
for(int i=0;i<nb_chauffeurs;i++)
```

```
{
```

```
    travaux_demandes[i] = IloNumArray2(env,nb_jours);
```

```
    for(int j=0;j<nb_jours;j++)
```

```
    {
```

```
        travaux_demandes[i][j]=IloNumArray(env,2);
```

```
        for(int k=0;k<2;k++)
```

```
            travaux_demandes[i][j][k]=0;
```

```
    }
```

```
}
```

```
//Remplissage des cases jaunes ([conducteur-1][jour-1][creneau])
```

```
//creneau=0->matin creneau=1->soir
```

```
//REEMPLIR EN DUR LES CASES CONCERNEES
```

```
travaux_demandes[0][10][0]=1;
```

```
travaux_demandes[0][12][1]=1;
```

```
travaux_demandes[1][2][1]=1;
```

```
travaux_demandes[1][4][0]=1;
```

```
travaux_demandes[1][13][1]=1;
```

```
travaux_demandes[2][10][1]=1;
```

```
travaux_demandes[3][5][0]=1;
```

```
travaux_demandes[3][11][1]=1;
```

```
travaux_demandes[4][0][1]=1;
```

```
travaux_demandes[4][5][0]=1;
```

```
travaux_demandes[4][9][0]=1;
```

```
travaux_demandes[5][1][0]=1;
```

```
travaux_demandes[5][7][1]=1;
```

```
travaux_demandes[5][10][1]=1;
```

```
travaux_demandes[6][11][0]=1;
```

```
travaux_demandes[7][1][1]=1;
```

```
travaux_demandes[7][4][0]=1;
```

```
travaux_demandes[7][8][0]=1;
```

```
travaux_demandes[8][2][1]=1;
```

```
travaux_demandes[8][6][0]=1;
```

```
travaux_demandes[8][9][0]=1;
```

```
travaux_demandes[9][4][1]=1;
```

```
travaux_demandes[9][7][0]=1;
```

```
travaux_demandes[9][10][1]=1;
```

```
travaux_demandes[10][4][1]=1;
```

```
travaux_demandes[10][8][1]=1;
```

```
travaux_demandes[10][11][1]=1;
```

```
//creation de la table BigBlueBuses qui contiendra la planning de resolution du
probleme
```

```
//c'est la variable de descision
```

```
IloNumVarArray4 BigBlueBuses(env,nb_chauffeurs);
```

```
for(int i=0;i<nb_chauffeurs;i++)
```

```
{
```

```
    BigBlueBuses[i]=IloNumVarArray3(env,nb_jours);
```

```
    for(int j=0;j<nb_jours;j++)
```

```
    {
```

```
        BigBlueBuses[i][j]=IloNumVarArray2(env,nb_lignes);
```

```
        for(int k=0;k<nb_lignes;k++)
```

```
        {
```

```
            BigBlueBuses[i][j][k]=IloNumVarArray(env,2,0,1,ILOINT);
```

```
            //cette variable descision dependra de la valeur de cette dimension 4
```

```
            //0 pour le matin ou 1 pour le soir
```

```
        }
```

```
    }
```

```
}
```

```

//////////////////////////////////OBJECTIF//////////////////////////////////

//objectif1
//Respecter la préférence d'une équipe +3
IloExpr Objectif1(env);
for(int i=0;i<nb_chauffeurs;i++)
    for(int j=0;j<nb_jours;j++)
        for(int k=0;k<nb_lignes;k++)
            for(int l=0;l<2;l++)

                Objectif1=Objectif1+BigBlueBuses[i][j][k][l]*travails_demandes[i][j][l];

//objectif2
//Respecter la préférence d'un jour de congés d'un conducteur +4
IloExpr Objectif2(env);
for(int i=0;i<nb_chauffeurs;i++)
    for(int j=0;j<nb_jours;j++)
    {
        IloExpr crenaux_jours(env);
        for(int k=0;k<nb_lignes;k++)
        {
            for(int l=0;l<2;l++)
                crenaux_jours=crenaux_jours+BigBlueBuses[i][j][k][l];
        }
        Objectif2=Objectif2+(crenaux_jours==0)*conges_demandes[i][j];
    }

//objectif3
//Respecter un congés prolongé (3 jours ou plus) +5 (pour chaque 3 jours ou plus)
IloExpr Objectif3(env); //congés prolongé
//objectif 3 non réalisé

//objectif4
//our chaque equipe pas affecté -20
IloExpr Objectif4(env);
for(int i=0;i<nb_jours;i++)
    for(int j=0;j<nb_lignes;j++)
        for(int k=0;k<2;k++)
        {
            IloExpr crenaux_affecte(env);
            for(int l=0;l<nb_chauffeurs;l++)
                crenaux_affecte=crenaux_affecte+BigBlueBuses[l][i][j][k];
            Objectif4=Objectif4-crenaux_affecte;
        }
Objectif4=Objectif4+(nb_jours*nb_lignes*2);

//objectif5
//Pour chaque equipe de l'après-midi suivie immédiatement par l'equipe du matin dans
le Planning d'equipe -30
IloExpr Objectif5(env);
for(int i=0;i<nb_chauffeurs;i++)
    for(int j=0;j<nb_jours-1;j++)
    {
        IloExpr crenaux_soir(env);
        IloExpr prochain_crenaux_matin(env);
        for(int k=0;k<nb_lignes;k++)
        {
            crenaux_soir=crenaux_soir+BigBlueBuses[i][j][k][1];
            prochain_crenaux_matin=prochain_crenaux_matin+BigBlueBuses[i][j+1][k][0];
        }
        Objectif5=Objectif5+crenaux_soir*prochain_crenaux_matin;
    }

//objectif6
//Pour chaque equipe après la troisieme équipe dans l'après-midi affectée
consécutivement -10
IloExpr Objectif6(env);
for(int i=0;i<nb_chauffeurs;i++)

```

```

    for(int j=0;j<nb_jours-4;j++)
    {
        IloExpr crenaux_soir_quatre_jours(env);
        for(int k=j;k<j+4;k++)
        {
            for(int l=0;l<nb_lignes;l++)

                crenaux_soir_quatre_jours=crenaux_soir_quatre_jours+BigBlueBuses[i][k][l][1];
        }
        Objectif6=Objectif6+(crenaux_soir_quatre_jours==4);
    }

//objectif7
//Pour chaque equipe dans l'apres-midi affecté, qui n'est pas egal à 4 equipes -8
IloExpr Objectif7(env);
for(int i=0;i<nb_chauffeurs;i++)
{
    IloExpr crenaux_soir(env);
    for(int j=0;j<nb_jours;j++)
        for(int k=0;k<nb_lignes;k++)
            crenaux_soir=crenaux_soir+BigBlueBuses[i][j][k][1];
    Objectif7=Objectif7+IloAbs(4-crenaux_soir);
}

//fonction objectif

model.add(IloMaximize(env,3*Objectif1+4*Objectif2-20*Objectif4-30*Objectif5-10*Objectif6-8*Objectif7));

////////////////////////////////////////CONTRAINTES////////////////////////////////////////

//Les cases de congés(noires) ne peuvent etre utilisés
//Le chauffeur ne peut faire qu'une seule tournée
//Le conducteur ne peut travailler que sur ses lignes qui lui sont affectées
for(int i=0;i<nb_chauffeurs;i++)
{
    for(int j=0; j<nb_jours;j++)
    {
        IloExpr crenaux_par_jours(env);

        for(int k=0;k<nb_lignes;k++)
        {
            for(int l=0;l<2;l++)
            {

                model.add(BigBlueBuses[i][j][k][l]<=(disponibilite_conducteur[i][j]*affectation_lignes[i][k]));
                crenaux_par_jours=crenaux_par_jours+BigBlueBuses[i][j][k][l];
            }
        }
        model.add(crenaux_par_jours<=1);
    }
}

//Chaque crenau ne peut etre utilisé qu'une seule et unique fois
for(int i=0;i<nb_jours;i++)
{
    for(int j=0;j<nb_lignes;j++)
    {
        for(int k=0;k<2;k++)
        {
            IloExpr crenaux_affectes(env);
            for(int l=0;l<nb_chauffeurs;l++)
            {
                crenaux_affectes=crenaux_affectes+BigBlueBuses[l][i][j][k];
            }
            model.add(crenaux_affectes<=1);
        }
    }
}
}

```

```
//////////////////////////////////////RESOLUTION//////////////////////////////////////
```

```
IloCplex cplex(model);
```

```
if(cplex.solve())
```

```
{
```

```
    cplex.solve();
```

```
    IloNumArray2 table_fin(env,nb_chauffeurs);
```

```
    for(int i=0;i<nb_chauffeurs;i++)
```

```
        table_fin[i]=IloNumArray(env,nb_jours);
```

```
    env.out()<<"Score: "<<cplex.getObjValue()<<std::endl;
```

```
    for(int i=0;i<nb_chauffeurs;i++)
```

```
    {
```

```
        for(int j=0;j<nb_jours;j++)
```

```
        {
```

```
            for(int k=0;k<nb_lignes;k++)
```

```
            {
```

```
                for(int l=0;l<2;l++)
```

```
                {
```

```
                    if(cplex.getValue(BigBlueBuses[i][j][k][l])==1)
```

```
                    {
```

```
                        if(l==0)
```

```
                            table_fin[i][j]=-1*(k+1);
```

```
                        else
```

```
                            table_fin[i][j]=1*(k+1);
```

```
                    }
```

```
                }
```

```
            }
```

```
        }
```

```
    }
```

```
//////////////////////////////////////AFFICHAGE//////////////////////////////////////
```

```
//Le numero correspond a la ligne, le zero correspond a aucun crenau a placer.
```

```
//L'affichage vert correspond a la ligne 1
```

```
//L'affichage jaune correspond a la ligne 2
```

```
//L'affichage rouge correspond a la ligne 3
```

```
//Si le nombre est negatif le crenau est le matin, si il est positif le crenau est le soir
```

```
for(int i=0;i<nb_chauffeurs;i++)
```

```
{
```

```
    if(i<9)
```

```
        std::cout<<"Chauffeur "<<i+1<<": ";
```

```
    else
```

```
        std::cout<<"Chauffeur "<<i+1<<": ";
```

```
    for(int j=0;j<nb_jours;j++)
```

```
    {
```

```
        if(table_fin[i][j]<0)
```

```
        {
```

```
            if(table_fin[i][j]==-1)
```

```
                std::cout<<"\E[32;1m"<<table_fin[i][j]<<"\E[m ";
```

```
            else if(table_fin[i][j]==-2)
```

```
                std::cout<<"\E[33;1m"<<table_fin[i][j]<<"\E[m ";
```

```
            else if(table_fin[i][j]==-3)
```

```
                std::cout<<"\E[31;1m"<<table_fin[i][j]<<"\E[m ";
```

```
        }
```

```
        else if(table_fin[i][j]>0)
```

```
        {
```

```
            if(table_fin[i][j]==1)
```

```
                std::cout<<"\E[32;1m"<<table_fin[i][j]<<"\E[m ";
```

```
            else if(table_fin[i][j]==2)
```

```
                std::cout<<"\E[33;1m"<<table_fin[i][j]<<"\E[m ";
```

```
            else if(table_fin[i][j]==3)
```

```
                std::cout<<"\E[31;1m"<<table_fin[i][j]<<"\E[m ";
```

```
        }
```

```
    else
```

```
        std::cout<<table_fin[i][j]<<" ";
```

```
        }
        std::cout<<std::endl;
    }
}

catch(IloException exception)
{
    env.out()<<"Il y a un probleme"<<endl;
    env.out()<<exception<<endl;
}
//destruction de l'environnement
env.end();
return 0;
}
```