

	<h1>Miniprojet Janken</h1>
SPE - ISN	Elaborer un mini projet en C
Mini projet	
Organisation : Travail maison	

Algorithme du moteur

Algorithme moteur

Les arguments sont :

- joueur : entier
- ordi : entier

resultat : chaîne ;
joueurGagne booléen;

DEBUT

joueurGagne=faux ;

SELON QUE ordi est égal à

1 : resultat= « L'ordinateur joue les ciseaux »

2 : resultat= « L'ordinateur joue la feuille »

3 : resultat= « L'ordinateur joue la pierre »

FIN SELON

SI joueur=ordi

ALORS

Resultat=resultat+ « Match nul ! »

SINON

SELON QUE joueur est égal à

1 : **SI** (ordi=2) **ALORS** joueurGagne=vrai ;

2 : **SI** (ordi=3) **ALORS** joueurGagne=vrai ;

3 : **SI** (ordi=1) **ALORS** joueurGagne=vrai ;

FIN SELON

SI (joueurGagne)

ALORS

Score=score+1

Resultat=result+ « Vous gagnez »

SINON

Score=score-1 ;

Resultat=result+ « Vous perdez »

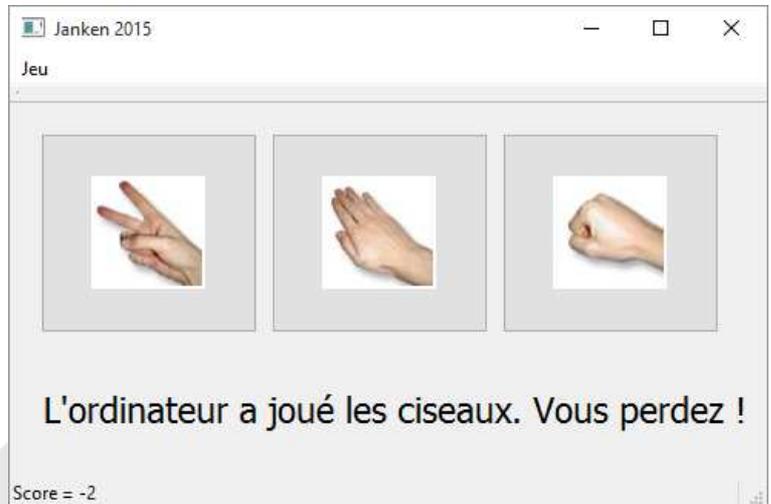
FINSI

FINSI

Ecrire resultat

Ecrire score;

FIN



CODE C

Mainwindow.h

```
#ifndef MAINWINDOW_H
#define MAINWINDOW_H

#include <QMainWindow>
#include <QLabel>

namespace Ui {
class MainWindow;
}

class MainWindow : public QMainWindow
{
    Q_OBJECT

public:
    explicit MainWindow(QWidget *parent = 0);
```

```

~MainWindow();
QLabel *statusLbl;
int score=0;
int choixOrdi();
void moteur(int ordi, int joueur);
public slots:
void on_ciseauxBtn_clicked();
void on_feuilleBtn_clicked();
void on_pierreBtn_clicked();
void on_actionNouveau_triggered();

private:
    Ui::MainWindow *ui;
};

#endif // MAINWINDOW_H

```

Mainwindow.cpp

```

#include "mainwindow.h"
#include "ui_mainwindow.h"
#include <time.h>

MainWindow::MainWindow(QWidget *parent) :
    QMainWindow(parent),
    ui(new Ui::MainWindow)
{
    ui->setupUi(this);
    statusLbl = new QLabel(); //créé le Label
    statusBar()->addWidget( statusLbl, 1 );
    statusLbl->setText("Score = 0");
    srand(time(NULL)); //on initialize le générateur
}

MainWindow::~MainWindow()
{
    delete ui;
}

//-----
void MainWindow::on_ciseauxBtn_clicked()
{ //l'utilisateur à choisi les ciseaux : choix 1
    moteur(1,choixOrdi());
}
//-----
void MainWindow::on_feuilleBtn_clicked()
{ //l'utilisateur à choisi la feuille : choix 2
    moteur(2,choixOrdi());
}
//-----
void MainWindow::on_pierreBtn_clicked()
{ //l'utilisateur à choisi la pierre : choix 3
    moteur(3,choixOrdi());
}
//-----
int MainWindow::choixOrdi()
{ //rend une valeur entre 1 et 3
    return (rand() % 3) + 1;
}
//-----
void MainWindow::moteur(int joueur,int ordi)
{
    QString resultat;
    bool joueurGagne=false;
    switch (ordi)

```

```

{
    case 1 : resultat="L'ordinateur a joué les ciseaux";break;
    case 2 : resultat="L'ordinateur a joué la feuille";break;
    case 3 : resultat="L'ordinateur a joué la pierre";break;
}
if (ordi==joueur)
{
    resultat=resultat+". Match nul !";
}
else
{ //pas de match nul
    switch (joueur)
    {
        case 1: if (ordi==2)joueurGagne=true;break;
        case 2:if (ordi==3)joueurGagne=true;break;
        case 3:if (ordi==1)joueurGagne=true;break;
    }
    if (joueurGagne==true)
    {
        score++;
        resultat=resultat+". Vous gagnez !";
    } else
    {
        resultat=resultat+". Vous perdez !";
        score=score-1;
    }
}
}
statusLbl->setText("Score = "+QString::number(score));
ui->texte->setText(resultat);
}
//-----
void MainWindow::on_actionNouveau_triggered()
{ //on démarre un nouveau jeu
    score=0;
    statusLbl->setText("Score = "+QString::number(score));
    ui->texte->setText("Appuyez sur un bouton !");
}

```

Main.cpp

```

#include "mainwindow.h"
#include <QApplication>

int main(int argc, char *argv[])
{
    QApplication a(argc, argv);
    MainWindow w;
    w.show();

    return a.exec();
}

```

Remarques :

- Observer comment est généré le nombre aléatoire
- Observer comment on modifie la barre d'état