

# ENIGMA

**Alumne:** Jordi Ribellas Ramos

**Professor:** Francis Real

**Curs:** 2022/23

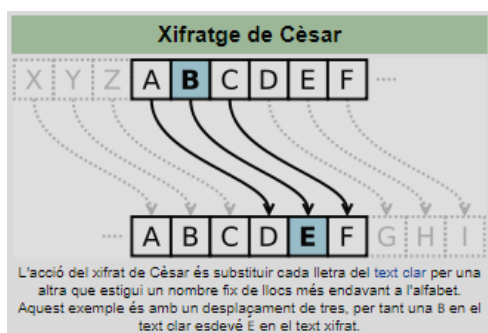
## 1- Enunciat

- o En aquesta activitat es demana:

La pel·lícula "Desxifrant a Enigma" narra com Alan Turing i el seu equip de desxifrat de codis intenta de trencar el xifrat de la màquina Enigma de l'Alemanya Nazi durant els dies més foscos de la Segona Guerra Mundial.

En aquesta activitat crearem un programa per xifrar i desxifrar missatges simples emulant la màquina enigma.

La nostra màquina estarà basada en el xifrat de Cèsar. En aquest xifrat cada lletra se substitueix per una altra lletra desplaçada n espais en l'alfabet.



Per al xifrat farem servir 3 codificacions, cadascuna representada per un desplaçament: {x, y, z}.

D'aquesta manera, la primera lletra estarà desplaçada x posicions, la segona y posicions, la tercera z posicions, la quarta x posicions, ... i així successivament.

Farem servir només lletres majúscules sense espais. Es substituiran en el missatge caràcters especials dels idiomes Ñ->N o Ç-> C

Farem servir només les lletres: { A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z} en aquest ordre. Els desplaçaments seran nombres del 0 al 25.

Per exemple, si volem codificar el títol de la pel·lícula "End Game" amb la codificació {1,2,3} primer preparariem el text posant-en majúscules i sense espais: "ENDGAME" (l'usuari ho prepara) i la nostra màquina ho codificaria en "FPGHCPF".

En aquesta activitat es demana:

- Una finestra que contingui 5 quadres de text:
  - Text a codificar (Només majúscules i sense espais)
  - 3 per als desplaçaments {x, y, z}
  - Text codificat (només de sortida, no modificable per l'usuari)
- Etiquetes per titular cada quadre de text.
- Botó per iniciar la codificació.
- Botó per iniciar la descodificació.

Es valorarà la presentació de la finestra (colors, justificació, ús de tipus de lletra, etc ...).

Es demana també utilitzar el programa per descodificar el següent missatge:

"HTDRJYCFOKWPOSFJHLAXEWWLJYLJZYEIZOHZITWWQTCFOAQLFZDPNNEFAKWBJDQXXWIC  
AXYKXPHQLIFYIFCPF"

usant els desplaçaments {4, 21, 15}



## 2- Codi

- o En aquest apartat es veurà el meu codi.

```
import java.awt.Container;
import java.awt.FlowLayout;
import java.awt.GridLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JTextField;
import javax.swing.SwingConstants;

public class codificacionEnigma extends JFrame {

    JPanel original = new JPanel();
    JPanel codificat = new JPanel();
    JPanel botons = new JPanel();
    JPanel resultat = new JPanel();
    Container = getContentPane();

    JTextField entrada = new JTextField();
    JTextField sortida = new JTextField();

    JTextField codificacio1 = new JTextField(3);
    JTextField codificacio2 = new JTextField(3);
    JTextField codificacio3 = new JTextField(3);

    Boolean error = false;

    public codificacionEnigma() {

        super("Enigma");
        setSize(800, 300);

        GridLayout gl = new GridLayout(4, 1);

        // Assignem un Layout a cada Panel
        original.setLayout(new GridLayout(2, 1));
        codificat.setLayout(new FlowLayout());
        botons.setLayout(new FlowLayout());
        resultat.setLayout(new GridLayout(2, 1));

        // Afegim que volem que hi hagi a cada Panel
        JLabel label1 = new JLabel("Escriu el missatge:");
        label1.setHorizontalAlignment(SwingConstants.CENTER);
        original.add(label1);

        entrada.setHorizontalAlignment(SwingConstants.CENTER);
        original.add(entrada);

        JLabel text = new JLabel("Escriu els numeros que codificaran:");
```

```

text.setHorizontalAlignment(SwingConstants.CENTER);
codificat.add(text);

codificacio1.setHorizontalAlignment(SwingConstants.CENTER);
codificat.add(codificacio1);

codificacio2.setHorizontalAlignment(SwingConstants.CENTER);
codificat.add(codificacio2);

codificacio3.setHorizontalAlignment(SwingConstants.CENTER);
codificat.add(codificacio3);

JButton encriptar = new JButton("Encriptar");
encriptar.addActionListener(new encript());

JButton desencriptar = new JButton("Desencriptar");
desencriptar.addActionListener(new decript());

botons.add(encriptar);
botons.add(desencriptar);

JLabel label2 = new JLabel("Aquest es el resultat");
label2.setHorizontalAlignment(SwingConstants.CENTER);
resultat.add(label2);

sortida.setHorizontalAlignment(SwingConstants.CENTER);
sortida.setEditable(false);
resultat.add(sortida);

// Ho afegim al container
container.setLayout(gl);
container.add(original);
container.add(codificat);
container.add(botons);
container.add(resultat);
}

public class encript implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        error = false;

        int cont = 1;
        String text = entrada.getText();
        char[] chars = text.toCharArray();
        char[] solucio = new char[chars.length];

        int num1 = Integer.parseInt(codificacio1.getText());
        int num2 = Integer.parseInt(codificacio2.getText());
        int num3 = Integer.parseInt(codificacio3.getText());

        int restador = 0, codi = 90, tope = 64;

        // cridar la funcio de comprovar caracters
        comprovarCaracters(chars);

        if (!error) {

```

```

for (int i = 0; i < text.length(); i++) {
    restador = 0;

    switch (cont) {
        case 1:
            int ascii1 = (int) chars[i];

            if ((ascii1 + num1) > 90) {
                restador = ascii1 + num1;
                restador = restador - codi;
                ascii1 = tope + restador;
            } else {
                ascii1 = ascii1 + num1;
            }
            solucio[i] = (char) ascii1;
            cont = 2;
            break;

        case 2:
            int ascii2 = (int) chars[i];

            if ((ascii2 + num2) > 90) {
                restador = ascii2 + num2;
                restador = restador - codi;
                ascii2 = tope + restador;
            } else {
                ascii2 = ascii2 + num2;
            }
            solucio[i] = (char) ascii2;
            cont = 3;
            break;

        case 3:
            int ascii3 = (int) chars[i];

            if ((ascii3 + num3) > 90) {
                restador = ascii3 + num3;
                restador = restador - codi;
                ascii3 = tope + restador;
            } else {
                ascii3 = ascii3 + num3;
            }
            solucio[i] = (char) ascii3;
            cont = 1;
            break;

        default:
            break;
    }
}

String str = new String(solucio);
sortida.setText(str);
}
}

```



```

        ascii3 = tope - restador;
    } else {
        ascii3 = ascii3 - num3;
    }
    solucio[i] = (char) ascii3;
    cont = 1;
    break;

    default:
        break;
    }
    }
}

String str = new String(solucio);
sortida.setText(str);

}

}

public void comprovarCaracters(char[] array) {

    Pattern = Pattern.compile("[A-Z]");
    String comprovarMajuscules = new String(array);
    if (entrada.getText().isEmpty()) {
        System.out.println("Error: El text d'entrada no pot estar
buit");
        error = true;
    } else if(!error){
        for (int i = 0; i < array.length; i++) {
            Matcher = pattern.matcher(comprovarMajuscules);
            boolean matchFound = matcher.find();

            if (!matchFound && error == false) {
                System.out.println("Error: El text d'entrada
han de ser lletres majuscules i no ha contindre numeros");
                error = true;
            }
            if (array[i] == ' ' && error == false) {
                System.out.println("Error: El text d'entrada
no pot tindre espais en blanc");
                error = true;
            }
            if (array[i] == 'Ñ') {
                array[i] = 'N';
            }
            if (array[i] == 'Ç') {
                array[i] = 'C';
            }
        }
    }
}

}

}
}

```



### 3- Documentació del codi:

- o En aquest apartat comentaré el codi funció per funció

3.1. El primer que podem veure al codi es la declaració dels JTextField que farem servir en el codi. Aquests son arrays ja que tractarem les dades com arrays y no com a variables individuals.

3.2. Després podem veure com declarem la nova finestra en la que treballarem i li donem les seves mides, sota d'això declararem tots els FlowLayouts per a que estiguin al mig i estiguin centrats, a sota crearem els botons, cadascun cridant a la seva funció corresponent, i ho afegirem tot a un container general.

#### 3.3. Funcions dels botons

##### a. Encript

- El que fem en aquesta funcio es convertir el text introduït a un array de chars, després recorrerem tota la longitud del text comprovant amb un switch quina lletra hem de mirar. En el cas de mirar la primera lletra, passarem el char a int i comprovarem que la lletra en codi ascii no sigui superior a 90, si no ho es sumarem a una altre variable complementaria el calor de la lletra en ascii i el valor del primer numero introduït. Després restarem aquest resultat a 90. I tenint aquest valor sumarem el resultat anterior a 64 per a que ens retorni la lletra corresponent. I passarem el valor del comptador per a que miri la segona lletra.

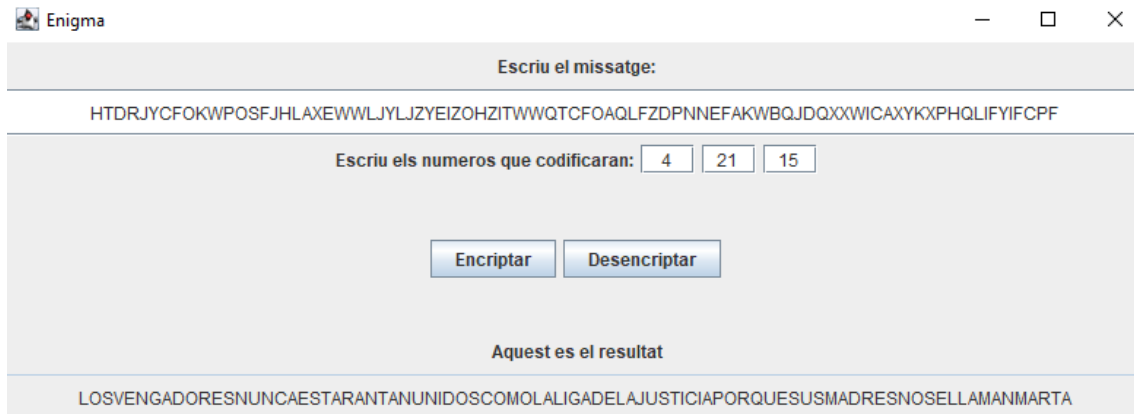
##### b. Decript

- En aquesta funció a comparació de la funció d'enciptar, farem el mateix per en lloc de sumar, restarem per a tirar endarrere en la posició del codi ascii.

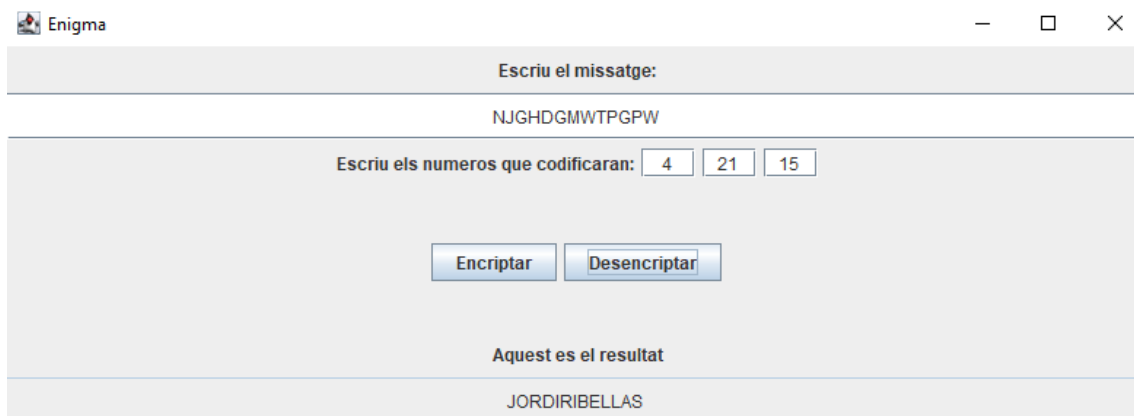
## 4- Proves

- Aquí farem les proves de que l'aplicació funciona correctament.

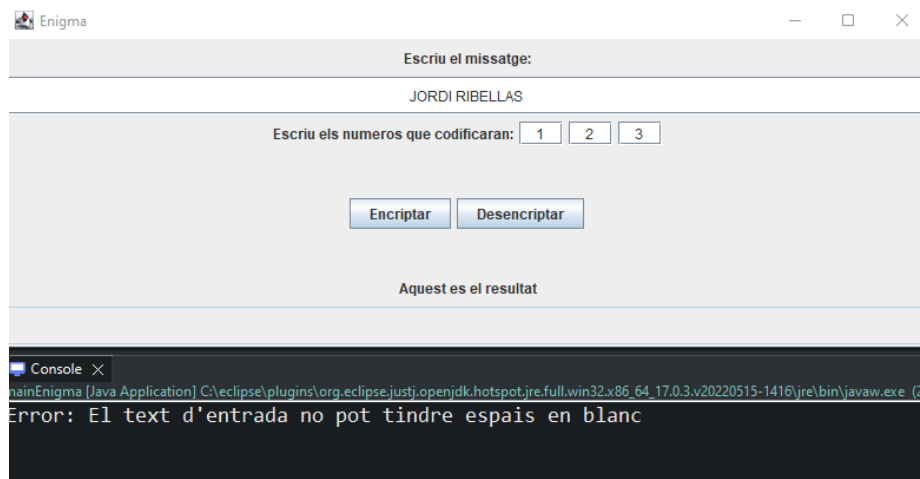
1) En aquesta captura de pantalla podem veure que el codi encripta correctament.



2) En aquesta captura de pantalla podem veure com desencripta correctament



3) Aquí podem veure que si tenim espais en blanc ens ho dirà



- 4) En aquesta captura podem veure que si posem lletres en minúscula no ens funcionarà

