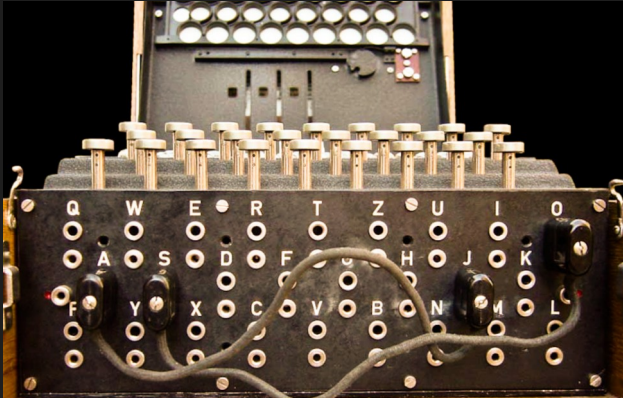


## Captures d'écran et illustrations Enigma

### 1. Illustration des fonctions par rapport au mécanisme réel



#### Initialisation

cablageInitial

choixRotor  
choixReflecteur  
positionInitialeRotor

#### Execution

valeurApresCablageDeDepart

passageDansUnRotor  
passageDansLeReflecteur  
inverseRotor  
decalageUnRang

### 2. Fonctions

```
78 // Exemples :
79 // entree : 'A' -> sortie : "YRUHQSLDPXNGOKMIEBFZCWVJAT" (REFLECTEURA)
80 // entree : 'B' -> sortie : "RDOBJNTKVEHMLFCWZAXGYIPSUQ" (REFLECTEURB)
81 String choixReflecteur(char lettreReflecteur){
82     String res = "";
83     if (lettreReflecteur == 'A') {
84         res = REFLECTEURA;
85     } else if (lettreReflecteur == 'B'){
86         res = REFLECTEURB;
87     } else {
88         println("ERREUR : Les choix possibles sont A ou B.");
89     }
90     return res;
91 }
92 // -----
93 // Fonction qui permet a l'utilisateur de la machine de brancher les cables reliant les paires (6) de lettres
94 // Cette fonction doit retourner une chaine de caracteres de 6 lettres majuscules saisies au clavier par l'utilisateur (on supposera que c
95 // Exemple :
96 // Si l'utilisateur saisit les 6 paires suivantes : AV puis DE puis HO puis JK puis LS puis XQ, la fonction doit retourner "AVDEHOJKLSXQ"
97 String cablageInitial(){
98     println("Entrez les 6 paires du cablage initial");
99     String pair = "";
100     for (int cpt = 0; cpt<6; cpt++) {
101         pair = pair + enMajuscule(readString());
102     }
103     return pair;
104 }
105 void affichageCablageInitial(){
106     println(cablageInitial());
107 }
```

### 3. Programme principal

```
232 void algorithm(){
233     println(" -----\n| Simulation d'une machine Enigma M3 |\n -----");
234
235     // Initialisation des elements
236     // -----
237     println("Quel type de configuration souhaitez-vous ? \n 1 : Configuration par default (Rotor 1 : III en position W, Rotor 2 : I en position W, Rotor 3 : I en position W)
238     int choix = readInt();
239     while (choix<1 || choix>3){
240         println("Erreur de saisie, choix : 1, 2 ou 3");
241         choix = readInt();
242     }
243
244     if (choix == 1){// Configuration pre-etablie
245         // // Rotors choisis
246         rotor1 = 3;
247         rotor2 = 1;
248         rotor3 = 5;
249         R1=choixRotor(rotor1);
250         R2=choixRotor(rotor2);
251         R3=choixRotor(rotor3);
252
253         // // Position initiale des rotors choisis
254         position1 = 'W';
255         position2 = 'D';
256         position3 = 'E';
257         decalage1 = indiceLettre(position1,R1);
258         R1 = positionInitialeRotor(R1,decalage1);
259         decalage2 = indiceLettre(position2,R2);
260         R2 = positionInitialeRotor(R2,decalage2);
261         decalage3 = indiceLettre(position3,R3);
262         R3 = positionInitialeRotor(R3,decalage3);
263
264         // // Reflecteur choisi
265         choixRef = 'B';
266         refl = choixReflecteur(choixRef);
267
268         // // Initialisation de la configuration du cablage de la machine par l'utilisateur
269         cables = "AVDEHOJKLSXQ";
270
271         // // Message a tester
272         message = "AKBAOKETGPVYHGWBSGSVUDTZEBNXGFOBVOJVTWFPKIC";
273     }
274     else if (choix == 2){// Configuration choisie par l'utilisateur
275         // // Choix des 3 rotors (distincts) parmi les 5
276         // NB : On supposera que les numeros des 3 rotors sont bien compris entre 1 et 5 et qu'ils sont tous differents
277         println("Entrez le numero du premier rotor choisi (1, 2, 3, 4, 5)");
278         rotor1 = readInt();
279         println("Entrez le numero du deuxieme rotor choisi");
280         rotor2 = readInt();
```

### 4. Menu de test des fonctions

```
321     println(" -----\n| Tests de fonctions |\n -----");
322
323     String fonc = "";
324     println("écrivez une fonction pour la tester. Entrez \"stop\" pour arreter les tests, \"list\" pour obtenir la liste des fonctions a tester.");
325     do {
326         print("Fonction a tester : ");
327         fonc = readString();
328         if (equals(fonc,"lettreEnNombre")){ // lettreEnNombre
329             print("Entrez un caractere : ");
330             println("La fonction retourne : "+lettreEnNombre(readChar()));
331         }else if (equals(fonc,"nombreEnLettre")){ // nombreEnLettre
332             print("Entrez un entier : ");
333             println("La fonction retourne : "+nombreEnLettre(readInt()));
334         }else if (equals(fonc,"choixRotor")){ // choixRotor
335             print("Entrez un entier : ");
336             println("La fonction retourne : "+choixRotor(readInt()));
337         }else if (equals(fonc,"choixReflecteur")){ // choixReflecteur
338             print("Entrez un caractere : ");
339             println("La fonction retourne : "+choixReflecteur(readChar()));
340         }else if (equals(fonc,"cablageInitial")){ // cablageInitial
341             println("La fonction retourne : "+cablageInitial());
342         }else if (equals(fonc,"decalageUnRang")){ // decalageUnRang
343             print("Entrez une chaine de caracteres : ");
344             println("La fonction retourne : "+decalageUnRang(readString()));
345         }else if (equals(fonc,"positionInitialeRotor")){ // positionInitialeRotor
346             print("Entrez une chaine de caracteres : ");
347             String temp = readString();
348             print("Entrez un entier : ");
349             println("La fonction retourne : "+positionInitialeRotor(temp,readInt()));
350         }else if (equals(fonc,"indiceLettre")){ // indiceLettre
```

## 5. Boucle principale qui décrypte le message

```
386 String messageDecode="";
387 char lettre;
388 int cpt1 = 0;
389 int cpt2 = 0;
390 // Boucle principale du programme Enigma
391 // -----
392 for (int tour=0 ; tour < length(message) ; tour=tour+1){// la boucle s'arrete quand on a code chaque lettre du message
393     // 1. Recuperation de la lettre courante dans le message a decoder
394     lettre = charAt(message,tour);
395     // 2. Passage par le cablage
396     lettre = valeurApresCablageDeDepart(lettre, cables);
397     // 3. Passage par les 3 rotors (premier, deuxieme, troisieme)
398     lettre = passageDansUnRotor(lettre,R1);
399     lettre = passageDansUnRotor(lettre,R2);
400     lettre = passageDansUnRotor(lettre,R3);
401     // 4. Passage par le reflecteur
402     lettre = passageDansLeReflecteur(lettre,refl);
403     // 5. Passage par les 3 rotors (dans le sens inverse : troisieme, deuxieme, premier)
404     lettre = inverseRotor(lettre,R3);
405     lettre = inverseRotor(lettre,R2);
406     lettre = inverseRotor(lettre,R1);
407     // 6. Passage par le cablage
408     lettre = valeurApresCablageDeDepart(lettre, cables);
409
410     // 7. Ajout de la lettre decodee au message
411     messageDecode = messageDecode + lettre;
412
413     // 8. Preparation de l'iteration suivante : decalage du premier rotor (a chaque fois); decalage du deuxieme rotor si le premier a
414     // 8.1 Le rotor 1 tourne d'un rang vers la gauche apres chaque lettre (donc a chaque tour)
415     R1 = decalageUnRang(R1);
416     cpt1 += 1;
417     // 8.2 Si le rotor 1 a effectue un tour (toutes les 26 iterations) alors le rotor 2 tourne d'un cran vers la gauche
418     if (cpt1 == 26){
419         R2 = decalageUnRang(R2);
420         cpt1 = 0;
421         cpt2 += 1;
422     }
423     // 8.3 Si le rotor 2 a effectue un tour (toutes les 26*26 iterations) alors le rotor 3 tourne d'un cran vers la gauche
424     if (cpt2 == 26){
425         R3 = decalageUnRang(R3);
426         cpt2 = 0;
427     }
428 }
429 println("Le message decode est : \n" + messageDecode);
430 }
431 }
```