```
/*-----------------------------------------------------------------------------------------------
 * Author:      Dan Cassidy
 * Date:        2015-08-04
 * Assignment:  Project
 * Source File: RotorType.java
 * Language:    Java
 * Course:      CSCI-C 490, Android Programming, MoWe 08:00
-----------------------------------------------------------------------------------------------*/
package com.chaoticcognitions.aenigma.models.rotors;

/**
 * Enum to store the relevant information about the different types of rotors in a single place.
 * @author Dan Cassidy
 */
public enum RotorType {
    // Enigma I - German Army and Air Force (Wehrmacht, Luftwaffe)
    // Stator
    I_ETW,
    // Rotors
    I_I, I_II, I_III, I_IV, I_V,
    // Reflectors
    I_UKW_A, I_UKW_B, I_UKW_C,

    // Norway Enigma - Enigma I used postwar by Norway
    // Stator
    N_ETW,
    // Rotors
    N_I, N_II, N_III, N_IV, N_V,
    // Reflectors
    N_UKW,

    // Enigma M3 - German Navy (Kriegsmarine)
    // Stator
    M3_ETW,
    // Rotors
    M3_I, M3_II, M3_III, M3_IV, M3_V, M3_VI, M3_VII, M3_VIII,
    // Reflectors
    M3_UKW_B, M3_UKW_C,

    // Enigma M4 - U-Boot Enigma
    // Stator
    M4_ETW,
    // Rotors
    M4_I, M4_II, M4_III, M4_IV, M4_V, M4_VI, M4_VII, M4_VIII, M4_BETA, M4_GAMMA,
    // Reflectors
    M4_UKW_B, M4_UKW_C,

    // Enigma G - Zählwerk Enigma A28 and G31
    // Stator
    G_ETW,
    // Rotors
    G_I, G_II, G_III,
    // Reflectors
    G_UKW,

    // Enigma D - Commercial Enigma A26
    // Stator
    D_ETW,
    // Rotors
    D_I, D_II, D_III,
```

```
61        // Reflectors
62        D_UKW,
63
64        // Enigma K - Commercial Enigma A27
65        // Stator
66        K_ETW,
67        // Rotors
68        K_I, K_II, K_III,
69        // Reflectors
70        K_UKW,
71
72        // Swiss K - Swiss Enigma K Variant (Swiss Air Force)
73        // Stator
74        KS_ETW,
75        // Rotors
76        KS_I, KS_II, KS_III,
77        // Reflectors
78        KS_UKW,
79
80        // Enigma KD - Enigma K with UWK-D *** (Rewirable) *** //TODO figure out what to do with the KD
          enigma
81        // Stator
82        KD_ETW,
83        // Rotors
84        KD_I, KD_II, KD_III,
85        // Reflectors
86        KD_UKW,
87
88        // Railway Enigma - Modified Enigma K
89        // Stator
90        R_ETW,
91        // Rotors
92        R_I, R_II, R_III,
93        // Reflectors
94        R_UKW,
95
96        // Enigma T - Japanese Enigma (Tirpitz)
97        // Stator
98        T_ETW,
99        // Rotors
100       T_I, T_II, T_III, T_IV, T_V, T_VI, T_VII, T_VIII,
101       // Reflectors
102       T_UKW;
103
104       /**
105        * Get the wiring for the rotor based on its type. For instance, 'A' is wired to the first
106        * letter of this string, 'B' is wired to the second letter, and so on.
107        * @return The wiring for the rotor.
108        */
109       public String wiring() {
110           switch (this) {
111               // Stators
112               case I_ETW:
113               case N_ETW:
114               case M3_ETW:
115               case M4_ETW:
116                   return "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
117               case G_ETW:
118               case D_ETW:
119               case K_ETW:
```

```
120             case KS_ETW:
121             case KD_ETW:
122             case R_ETW:
123                 return "QWERTZUIOASDFGHJKPYXCVBNML";
124             case T_ETW:
125                 return "KZROUQHYAIGBLWVSTDXFPNMCJE";
126
127             // Rotors
128             case I_I:
129             case M3_I:
130             case M4_I:
131                 return "EKMFLGDQVZNTOWYHXUSPAIBRCJ";
132             case I_II:
133             case M3_II:
134             case M4_II:
135                 return "AJDKSIRUXBLHWTMCQGZNPYFVOE";
136             case I_III:
137             case M3_III:
138             case M4_III:
139                 return "BDFHJLCPRTXVZNYEIWGAKMUSQO";
140             case I_IV:
141             case N_IV:
142             case M3_IV:
143             case M4_IV:
144                 return "ESOVPZJAYQUIRHXLNFTGKDCMWB";
145             case I_V:
146             case M3_V:
147             case M4_V:
148                 return "VZBRGITYUPSDNHLXAWMJQOFECK";
149             case N_I:
150                 return "WTOKASUYVRBXJHQCPZEFMDINLG";
151             case N_II:
152                 return "GJLPUBSWEMCTQVHXAOFZDRKYNI";
153             case N_III:
154                 return "JWFMHNBPUSDYTIXVZGRQLAOEKC";
155             case N_V:
156                 return "HEJXQOTZBVFDASCILWPGYNMURK";
157             case M3_VI:
158             case M4_VI:
159                 return "JPGVOUMFYQBENHZRDKASXLICTW";
160             case M3_VII:
161             case M4_VII:
162                 return "NZJHGRCXMYSWBOUFAIVLPEKQDT";
163             case M3_VIII:
164             case M4_VIII:
165                 return "FKQHTLXOCBJSPDZRAMEWNIUYGV";
166             case M4_BETA:
167                 return "LEYJVCNIXWPBQMDRTAKZGFUHOS";
168             case M4_GAMMA:
169                 return "FSOKANUERHMBTIYCWLQPZXVGJD";
170             case G_I:
171             case D_I:
172             case K_I:
173                 return "LPGSZMHAEOQKVXRFYBUTNICJDW";
174             case G_II:
175             case D_II:
176             case K_II:
177                 return "SLVGBTFXJQOHEWIRZYAMKPCNDU";
178             case G_III:
179             case D_III:
```

```
180             case K_III:
181                 return "CJGDPSHKTURAWZXFMYNQOBVLIE";
182             case KS_I:
183                 return "PEZUOHXSCVFMTBGLRINQJWAYDK";
184             case KS_II:
185                 return "ZOUESYDKFWPCIQXHMVBLGNJRAT";
186             case KS_III:
187                 return "EHRVXGAOBQUSIMZFLYNWKTPDJC";
188             case KD_I:
189                 return "VEZIOJCXKYDUNTWAPLQGBHSFMR";
190             case KD_II:
191                 return "HGRBSJZETDLVPMQYCXAOKINFUW";
192             case KD_III:
193                 return "NWLHXGRBYOJSAZDVTPKFQMEUIC";
194             case R_I:
195                 return "JGDQOXUSCAMIFRVTPNEWKBLZYH";
196             case R_II:
197                 return "NTZPSFBOKMWRCJDIVLAEYUXHGQ";
198             case R_III:
199                 return "JVIUBHTCDYAKEQZPOSGXNRMWFL";
200             case T_I:
201                 return "KPTYUELOCVGRFQDANJMBSWHZXI";
202             case T_II:
203                 return "UPHZLWEQMTDJXCAKSOIGVBYFNR";
204             case T_III:
205                 return "QUDLYRFEKONVZAXWHMGPJBSICT";
206             case T_IV:
207                 return "CIWTBKXNRESPFLYDAGVHQUOJZM";
208             case T_V:
209                 return "UAXGISNJBVERDYLFZWTPCKOHMQ";
210             case T_VI:
211                 return "XFUZGALVHCNYSEWQTDMRBKPIOJ";
212             case T_VII:
213                 return "BJVFTXPLNAYOZIKWGDQERUCHSM";
214             case T_VIII:
215                 return "YMTPNZHWKODAJXELUQVGCBISFR";
216
217             // Reflectors
218             case I_UKW_A:
219                 return "EJMZALYXVBWFCRQUONTSPIKHGD";
220             case I_UKW_B:
221             case M3_UKW_B:
222                 return "YRUHQSLDPXNGOKMIEBFZCWVJAT";
223             case I_UKW_C:
224             case M3_UKW_C:
225                 return "FVPJIAOYEDRZXWGCTKUQSBNMHL";
226             case N_UKW:
227                 return "MOWJYPUXNDSRAIBFVLKZGQCHET";
228             case M4_UKW_B:
229                 return "ENKQAUYWJICOPBLMDXZVFTHRGS";
230             case M4_UKW_C:
231                 return "RDOBJNTKVEHMLFCWZAXGYIPSUQ";
232             case G_UKW:
233             case D_UKW:
234             case K_UKW:
235             case KS_UKW:
236                 return "IMETCGFRAYSQBZXWLHKDVUPOJN";
237             case KD_UKW:
238                 return "NSUOMKLIHZFGEADVXWBYCPRQTJ"; // Rewireable!
239             case R_UKW:
```

```
240                 return "QYHOGNECVPUZTFDJAXWMKISRBL";
241             case T_UKW:
242                 return "GEKPBTAUMOCNILJDXZYFHWVQSR";
243 
244             default:
245                 return "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
246         }
247     }
248 
249     /**
250      * Get the reverse wiring for the different types of rotors.
251      * @return
252      */
253     public String reverseWiring() {
254         String wiring = this.wiring();
255         char[] reverseWiring = new char[wiring.length()];
256 
257         final char CHAR_OFFSET = 'A';
258         for (int index = 0; index < wiring.length(); index++)
259             reverseWiring[wiring.charAt(index) - CHAR_OFFSET] = (char)(index + CHAR_OFFSET);
260 
261         return new String(reverseWiring);
262     }
263 
264     /**
265      * Get the turnover characters for the different types of rotors.
266      */
267     public String turnoverChars() {
268         switch (this) {
269             // Stators
270             case I_ETW:
271             case N_ETW:
272             case M3_ETW:
273             case M4_ETW:
274             case G_ETW:
275             case D_ETW:
276             case K_ETW:
277             case KS_ETW:
278             case KD_ETW:
279             case R_ETW:
280             case T_ETW:
281                 return "";
282 
283             // Rotors
284             case I_I:
285             case N_I:
286             case M3_I:
287             case M4_I:
288                 return "Q";
289             case I_II:
290             case N_II:
291             case M3_II:
292             case M4_II:
293                 return "E";
294             case I_III:
295             case N_III:
296             case M3_III:
297             case M4_III:
298                 return "V";
299             case I_IV:
```

```
300             case N_IV:
301             case M3_IV:
302             case M4_IV:
303                 return "J";
304             case I_V:
305             case N_V:
306             case M3_V:
307             case M4_V:
308                 return "Z";
309             case M3_VI:
310             case M4_VI:
311             case M3_VII:
312             case M4_VII:
313             case M3_VIII:
314             case M4_VIII:
315                 return "ZM";
316             case M4_BETA:
317             case M4_GAMMA:
318                 return "";
319             case G_I:
320                 return "SUVWZABCEFGIKLOPQ";
321             case G_II:
322                 return "STVYZACDFGHKMNQ";
323             case G_III:
324                 return "UWXAEFHKMNR";
325             case D_I:
326             case K_I:
327             case KS_I:
328             case R_III:
329                 return "Y";
330             case KS_II:
331             case D_II:
332             case K_II:
333             case R_II:
334                 return "E";
335             case D_III:
336             case K_III:
337             case KS_III:
338             case R_I:
339                 return "N";
340             case KD_I:
341             case KD_II:
342             case KD_III:
343                 return "SUYAEHLNQ";
344             case T_I:
345             case T_III:
346                 return "WZEKQ";
347             case T_II:
348             case T_IV:
349                 return "WZFLR";
350             case T_V:
351             case T_VII:
352                 return "YCFKR";
353             case T_VI:
354             case T_VIII:
355                 return "XEIMQ";
356
357             // Reflectors
358             case I_UKW_A:
359             case I_UKW_B:
```

```
360             case I_UKW_C:
361             case N_UKW:
362             case M3_UKW_B:
363             case M3_UKW_C:
364             case M4_UKW_B:
365             case M4_UKW_C:
366             case G_UKW:
367             case D_UKW:
368             case K_UKW:
369             case KS_UKW:
370             case KD_UKW:
371             case R_UKW:
372             case T_UKW:
373                 return "";
374
375             default:
376                 return "";
377         }
378     }
379
380     /**
381      * Get whether the rotor steps or not based on the type.
382      */
383     public boolean isSteppingRotor() {
384         switch (this) {
385             // Stators
386             case I_ETW:
387             case N_ETW:
388             case M3_ETW:
389             case M4_ETW:
390             case G_ETW:
391             case D_ETW:
392             case K_ETW:
393             case KS_ETW:
394             case KD_ETW:
395             case R_ETW:
396             case T_ETW:
397                 return false;
398
399             // Rotors
400             case I_I:
401             case I_II:
402             case I_III:
403             case I_IV:
404             case I_V:
405             case N_I:
406             case N_II:
407             case N_III:
408             case N_IV:
409             case N_V:
410             case M3_I:
411             case M3_II:
412             case M3_III:
413             case M3_IV:
414             case M3_V:
415             case M3_VI:
416             case M3_VII:
417             case M3_VIII:
418             case M4_I:
419             case M4_II:
```

```
420             case M4_III:
421             case M4_IV:
422             case M4_V:
423             case M4_VI:
424             case M4_VII:
425             case M4_VIII:
426             case G_I:
427             case G_II:
428             case G_III:
429             case D_I:
430             case D_II:
431             case D_III:
432             case K_I:
433             case K_II:
434             case K_III:
435             case KS_I:
436             case KS_II:
437             case KS_III:
438             case KD_I:
439             case KD_II:
440             case KD_III:
441             case R_I:
442             case R_II:
443             case R_III:
444             case T_I:
445             case T_II:
446             case T_III:
447             case T_IV:
448             case T_V:
449             case T_VI:
450             case T_VII:
451             case T_VIII:
452                 return true;
453             case M4_BETA:
454             case M4_GAMMA:
455                 return false;
456
457             // Reflectors
458             case I_UKW_A:
459             case I_UKW_B:
460             case I_UKW_C:
461             case N_UKW:
462             case M3_UKW_B:
463             case M3_UKW_C:
464             case M4_UKW_B:
465             case M4_UKW_C:
466             case D_UKW:
467             case K_UKW:
468             case KS_UKW:
469             case KD_UKW:
470             case R_UKW:
471             case T_UKW:
472                 return false;
473             case G_UKW:
474                 return true;
475
476             default:
477                 return false;
478         }
479     }
```

```
480
481       /**
482        * Get whether the rotor is marked with numbers or not based on the type.
483        */
484       public boolean isMarkedWithNumbers() {
485           switch (this) {
486               // Stators
487               case I_ETW:
488               case N_ETW:
489               case M3_ETW:
490               case M4_ETW:
491               case G_ETW:
492               case D_ETW:
493               case K_ETW:
494               case KS_ETW:
495               case KD_ETW:
496               case R_ETW:
497               case T_ETW:
498                   return false;
499
500               // Rotors
501               case I_I:
502               case I_II:
503               case I_III:
504               case I_IV:
505               case I_V:
506               case N_I:
507               case N_II:
508               case N_III:
509               case N_IV:
510               case N_V:
511                   return true;
512               case M3_I:
513               case M3_II:
514               case M3_III:
515               case M3_IV:
516               case M3_V:
517               case M3_VI:
518               case M3_VII:
519               case M3_VIII:
520               case M4_I:
521               case M4_II:
522               case M4_III:
523               case M4_IV:
524               case M4_V:
525               case M4_VI:
526               case M4_VII:
527               case M4_VIII:
528               case M4_BETA:
529               case M4_GAMMA:
530               case G_I:
531               case G_II:
532               case G_III:
533               case D_I:
534               case D_II:
535               case D_III:
536               case K_I:
537               case K_II:
538               case K_III:
539               case KS_I:
```

```
            case KS_II:
            case KS_III:
            case KD_I:
            case KD_II:
            case KD_III:
            case R_I:
            case R_II:
            case R_III:
            case T_I:
            case T_II:
            case T_III:
            case T_IV:
            case T_V:
            case T_VI:
            case T_VII:
            case T_VIII:
                return false;

            // Reflectors
            case I_UKW_A:
            case I_UKW_B:
            case I_UKW_C:
            case N_UKW:
            case M3_UKW_B:
            case M3_UKW_C:
            case M4_UKW_B:
            case M4_UKW_C:
            case G_UKW:
            case D_UKW:
            case K_UKW:
            case KS_UKW:
            case KD_UKW:
            case R_UKW:
            case T_UKW:
                return false;

            default:
                return false;
        }
    }

    /**
     * Get whether the rotor is a stator or not.
     */
    public boolean isStator() {
        switch (this) {
            // Stators
            case I_ETW:
            case N_ETW:
            case M3_ETW:
            case M4_ETW:
            case G_ETW:
            case D_ETW:
            case K_ETW:
            case KS_ETW:
            case KD_ETW:
            case R_ETW:
            case T_ETW:
                return true;

```

```
600             // Rotors
601             case I_I:
602             case I_II:
603             case I_III:
604             case I_IV:
605             case I_V:
606             case N_I:
607             case N_II:
608             case N_III:
609             case N_IV:
610             case N_V:
611             case M3_I:
612             case M3_II:
613             case M3_III:
614             case M3_IV:
615             case M3_V:
616             case M3_VI:
617             case M3_VII:
618             case M3_VIII:
619             case M4_I:
620             case M4_II:
621             case M4_III:
622             case M4_IV:
623             case M4_V:
624             case M4_VI:
625             case M4_VII:
626             case M4_VIII:
627             case M4_BETA:
628             case M4_GAMMA:
629             case G_I:
630             case G_II:
631             case G_III:
632             case D_I:
633             case D_II:
634             case D_III:
635             case K_I:
636             case K_II:
637             case K_III:
638             case KS_I:
639             case KS_II:
640             case KS_III:
641             case KD_I:
642             case KD_II:
643             case KD_III:
644             case R_I:
645             case R_II:
646             case R_III:
647             case T_I:
648             case T_II:
649             case T_III:
650             case T_IV:
651             case T_V:
652             case T_VI:
653             case T_VII:
654             case T_VIII:
655                 return false;
656
657             // Reflectors
658             case I_UKW_A:
659             case I_UKW_B:
```

```
660             case I_UKW_C:
661             case N_UKW:
662             case M3_UKW_B:
663             case M3_UKW_C:
664             case M4_UKW_B:
665             case M4_UKW_C:
666             case G_UKW:
667             case D_UKW:
668             case K_UKW:
669             case KS_UKW:
670             case KD_UKW:
671             case R_UKW:
672             case T_UKW:
673                 return false;
674
675             default:
676                 return false;
677         }
678     }
679
680     /**
681      * Get whether the rotor is an actual rotor or not.
682      */
683     public boolean isRotor() {
684         switch (this) {
685             // Stators
686             case I_ETW:
687             case N_ETW:
688             case M3_ETW:
689             case M4_ETW:
690             case G_ETW:
691             case D_ETW:
692             case K_ETW:
693             case KS_ETW:
694             case KD_ETW:
695             case R_ETW:
696             case T_ETW:
697                 return false;
698
699             // Rotors
700             case I_I:
701             case I_II:
702             case I_III:
703             case I_IV:
704             case I_V:
705             case N_I:
706             case N_II:
707             case N_III:
708             case N_IV:
709             case N_V:
710             case M3_I:
711             case M3_II:
712             case M3_III:
713             case M3_IV:
714             case M3_V:
715             case M3_VI:
716             case M3_VII:
717             case M3_VIII:
718             case M4_I:
719             case M4_II:
```

```
            case M4_III:
            case M4_IV:
            case M4_V:
            case M4_VI:
            case M4_VII:
            case M4_VIII:
            case M4_BETA:
            case M4_GAMMA:
            case G_I:
            case G_II:
            case G_III:
            case D_I:
            case D_II:
            case D_III:
            case K_I:
            case K_II:
            case K_III:
            case KS_I:
            case KS_II:
            case KS_III:
            case KD_I:
            case KD_II:
            case KD_III:
            case R_I:
            case R_II:
            case R_III:
            case T_I:
            case T_II:
            case T_III:
            case T_IV:
            case T_V:
            case T_VI:
            case T_VII:
            case T_VIII:
                return true;

            // Reflectors
            case I_UKW_A:
            case I_UKW_B:
            case I_UKW_C:
            case N_UKW:
            case M3_UKW_B:
            case M3_UKW_C:
            case M4_UKW_B:
            case M4_UKW_C:
            case G_UKW:
            case D_UKW:
            case K_UKW:
            case KS_UKW:
            case KD_UKW:
            case R_UKW:
            case T_UKW:
                return false;

            default:
                return false;
        }
    }

    /**
```

```
780         * Get whether the rotor is a reflector or not.
781         */
782        public boolean isReflector() {
783            switch (this) {
784                // Stators
785                case I_ETW:
786                case N_ETW:
787                case M3_ETW:
788                case M4_ETW:
789                case G_ETW:
790                case D_ETW:
791                case K_ETW:
792                case KS_ETW:
793                case KD_ETW:
794                case R_ETW:
795                case T_ETW:
796                    return false;
797
798                // Rotors
799                case I_I:
800                case I_II:
801                case I_III:
802                case I_IV:
803                case I_V:
804                case N_I:
805                case N_II:
806                case N_III:
807                case N_IV:
808                case N_V:
809                case M3_I:
810                case M3_II:
811                case M3_III:
812                case M3_IV:
813                case M3_V:
814                case M3_VI:
815                case M3_VII:
816                case M3_VIII:
817                case M4_I:
818                case M4_II:
819                case M4_III:
820                case M4_IV:
821                case M4_V:
822                case M4_VI:
823                case M4_VII:
824                case M4_VIII:
825                case M4_BETA:
826                case M4_GAMMA:
827                case G_I:
828                case G_II:
829                case G_III:
830                case D_I:
831                case D_II:
832                case D_III:
833                case K_I:
834                case K_II:
835                case K_III:
836                case KS_I:
837                case KS_II:
838                case KS_III:
839                case KD_I:
```

```
840                 case KD_II:
841                 case KD_III:
842                 case R_I:
843                 case R_II:
844                 case R_III:
845                 case T_I:
846                 case T_II:
847                 case T_III:
848                 case T_IV:
849                 case T_V:
850                 case T_VI:
851                 case T_VII:
852                 case T_VIII:
853                     return false;
854
855                 // Reflectors
856                 case I_UKW_A:
857                 case I_UKW_B:
858                 case I_UKW_C:
859                 case N_UKW:
860                 case M3_UKW_B:
861                 case M3_UKW_C:
862                 case M4_UKW_B:
863                 case M4_UKW_C:
864                 case G_UKW:
865                 case D_UKW:
866                 case K_UKW:
867                 case KS_UKW:
868                 case KD_UKW:
869                 case R_UKW:
870                 case T_UKW:
871                     return true;
872
873                 default:
874                     return false;
875             }
876         }
877
878         /**
879          * Returns the string representation of the rotor type.
880          * @return The string representation of the rotor type.
881          */
882         @Override public String toString() {
883             return super.toString().substring(super.toString().indexOf('_') + 1).replace('_', '-');
884         }
885     }
886
```