

Nov 04, 05 12:54

RepeatFinder.java

Page 1/1

```

import java.util.regex.*;
public class RepeatFinder {

    public String dna =
5 "ATGGCGGAGTCTTCGGATAAAGCTCTATCGAGTTCGAGTACGCCAAGAGCGGGCGCCTCTTGCAAGAA
ATGCAGCGAGACACTCCCAAGGACTCGCTCCGGATGGCCATCATGGTGCAGTCGCCCATGTTTGATGGA
AAAGTCCACACTGGTACCACCTCTCTGCTTCTGGAAGGTGGGCCACTCCATCCGGCACCCTGACGTTG
AGGTGGATGGGTTCTCTGAGCTTCGGTGGGATGACCAGCAGAAAGTCAAGAAGACAGCGGCACTGGAG
GAGTGACAGGCAAAAGCGGATGGAATTTGGTAGCAAGGCAGAGAAAGACTCTGGGTGACTTTGCAGCAG
AGTATGCCAAGTCCAACAGAAGTACGTGCAAGGGGTGTATGGAGAAGATAGAAAAGGGCCAGGTGCGC
CTGTCCAAGAAGATGGTGGACCCGGAGAAGCCACAGCTAGGCATGATTGACCGCTGGTACCATCCAGGC
TGCTTTGTCAAGAACAGGGAGGAGCTGGGTTCCGGCCCGAGTACAGTGGCAGTCAAGGGCTTCA
GCCTCCTTGCTACAGAGGATAAAGAAGCCCTGAAGAAGCAGCTCCAGGAGTCAAGAGTGAAGGAAAGA
GAAAAGGCGATGAGGTGGATGGAGTGGATGAAGTGGCGAAGAAGAAATCTAAAAAAGAAAAAGACAAG
GATAGAAGCTTGA AAAAGCCCTAAAGGCTCAGAACGACCTGATCTGGAACATCAAGGACGAGTAAAG
AAAGTGTGTTCAACTAAGTACCTACTACTCTTCAACAAGCAGCAAGTCTTCCGCTTCTGGGGAGT
CGGCGATCTTGGACCGAGTAGCTGATGGCATGGTGTTCGGTGCCTCCTCCCTGCGAGGAATGTCTGGG
TCAGCTGGTCTTCAAGAGCGATGCCTATTACTGCACCTGGGGACGCTACTGCCTGGACCAAGTATGGTC
AAGACACAGACACCCCAAGGAGGAGTGGGTAACCCCAAGGAATTCGGAGAAATCTTAACTCCTAAG
AAATTGAAGGTTAAAAAGAGGACCGATATTTCCCCCAAGAAACAGCGCTCCGTGGCGGCCACGCTC
CCGCCCTCCACAGCCTCGGCTCTGCTGTGAACCTCTGCTTACAGCAGATAAGCCATTAAGCCAT
GAAGATCCTGACTCTCGGAAGCTGTCCCGGAACAAGGATGAAGTGAAGGCCATGATTGAGAACTCGG
GGGAAGTTGACGGGACGGCAAGCTAACAGCTTCCCTGTGCATCAGCACCAAAAAGGAGTGGAAAGA
TGAATAAGAAGATGGAGGAAGTAAAGGAAGCCAAACATCCGAGTTGTGTCTGAGGACTTCTCCAGGACG
TCTCCGCTCCACCAAGGCTTCAAGGATGTTCTTAGCCACATCTTGTCCCTTGGGGGGCAGAGGT
GAAGGCAGGCGCTTGAAGTTTGGAGGAGGATGGCCATTAAGCAGTTTATGAAGAAAAAAGCAA
GGGCCAGGTCAAGGAGGAAGGTATCAACAAATCTGAAAAGAGATAAAATTAATCTTAAAGGAGGAG
CAGCTGTGGTGAACACTTGGCAACTCTGCGCATGCTTGGAGAAAGGTGGGAGGTTCTCAG
TGCCACCCTTGGCCTGGTGGACATCGTTAAAGGAACCAACTCTACTACAAGTGCAGCTTCTGGAGGAC
GACAAGGAAAACAGGTATTGGATATTCAGTTCCTGGGGCCGTGTGGGTACGGTGATCGGTAGCAACAAA
CTGGAACAGATGGCGTCCAAGGAGGATGCCATTAAGCAGTTTATGAAGAAAAAAGCAA
AACGCTTGGCACTCCAAAAATTTACAGAAATTCACCAAAAGTTTTACCCCTGGAGATTGACTATGGCC
AGGATGAAGAGGCGAGTGAAGAAAGTCAAGTAAATCTGGCAACCAAGTCCAAGCTCCCAAGGCCAGTTC
AGGACCTCATCAAGATGATCTTTGATGTGGAAGATATGAAGAAAGCCATGGTGGAGTATGAGATCGACC
TTCAGAAGATGCCTTGGGAAGCTGAGCAAAAAGGCGAGTCCAGGCGCATACTCCATCCTCAGTGAGG
TCCAGCAGGCGGTGCTCAGGGCAGCAGGACTCTCAGATCTCTGGATCTCTCAAATCGCTTTTACACCCT
GATCCCCACGACTTTGGGATGAAGAAGCTCCGCTCCTGAACAATGCAGACAGTGTGCAAGGCCAAGGT
GGAAATGCTTGACAACCTGCTGGACATCGAGGTGGCTACAGTCTGCTCAGGGAGGGTGTGATGATAG
CAGCAAGGATCCCATCGATGCAACTATGAGAAGTCAAAAAGTACATTAAGGTGGTTCAGATGATGATG
GAAGAAGCCGAGATCATCGAAGATGTTAAGAACAACACTCAACACCACACACAGCTGCGTACTGCTG
GAAGTCATCGATATCTTAAAGATAGAGCGTGAAGGCGAATGCCAGCGTTACAAGCCCTTAAAGCAGCTTC
ATAACCGAAGATGCTGTGGCAGGGTCCAGGACCAACACTTGTCTGGGATCCTGTCCAGGGTCTTCG
GATAGCCCGCCTGAAGCGCCGTGACAGGATACATGTTGGTAAAGGGATCTATTTTCGCTGACATGGTC
TCCAAGATGCAACTACTACATACGCTCAGGGAGACCAATAGGCTTAATCCTGTTGGGAGAAGTTG
CCCTTGGAAACATGATGAACCTGAAGCAGGCTTACATATCAGCAGGTTACCAAGGGCAAGCAGGTG
CAAAGGTTTGGGCAAACTACCCCTGATCCTCAGCTAACATTAAGTCTGGATGGTGTAGACGTTCTCTTG
GGACCGGGATTTCACTGTTGATAGACACCTCTACTATATAACGAGTACATGTTCTATGATATTGCT
CAGCTAAATCTGAAGTATCTGCTGAAACTGAAATTCAAATTTAAAGACCTCCCTGTGGTAA";

    public void find(){
        //Pattern pat = Pattern.compile("(...).*?(\\1)");
        Pattern pat = Pattern.compile("(...)(\\1)");
        Matcher match = pat.matcher(dna);
10     while (match.find()){
            int first = match.start();
            int end = match.end();
            System.out.println(match.group(1));
            System.out.println(match.group(2));
            System.out.println(first + " to " + end);
        }
20     /**
     * @param args
     */
    public static void main(String[] args) {
        RepeatFinder rf = new RepeatFinder();
25     rf.find();
    }
}

```

Oct 24, 05 12:36

RepeatCount.java

Page 1/4

```

public class RepeatCount {

    private static String[] repeat1Array = {
5     "A", "C", "T", "G",
    };

    private static String[] repeat2Array = {
10     "AA", "CA", "TA", "GA",
    "AC", "CC", "TC", "GC",
    "AT", "CT", "TT", "GT",
    "AG", "CG", "TG", "GG",
    };

    private static String[] repeat3Array = {
15     "AAA", "CAA", "TAA", "GAA",
    "ACA", "CCA", "TCA", "GCA",
    "ATA", "CTA", "TTA", "GTA",
    "AGA", "CGA", "TGA", "GGA",
    "AAC", "CAC", "TAC", "GAC",
20     "ACC", "CCC", "TCC", "GCC",
    "ATC", "CTC", "TTC", "GTC",
    "AGC", "CGC", "TGC", "GGC",
    "AAT", "CAT", "TAT", "GAT",
    "ACT", "CCT", "TCT", "GCT",
25     "ATT", "CTT", "TTT", "GTT",
    "AGT", "CGT", "TGT", "GGT",
    "AAG", "CAG", "TAG", "GAG",
    "ACG", "CCG", "TCG", "GCG",
    "ATG", "CTG", "TTG", "GTG",
30     "AGG", "CGG", "TGG", "GGG",
    };

    private static String[] repeat4Array = {
35     "AAAA", "CAAA", "TAAA", "GAAA",
    "ACAA", "CCAA", "TCAA", "GCAA",
    "ATAA", "CTAA", "TTAA", "GTAA",
    "AGAA", "CGAA", "TGAA", "GGAA",
    "AACA", "CACA", "TACA", "GACA",
    "ACCA", "CCCA", "TCCA", "GCCA",
40     "ATCA", "CTCA", "TTCA", "GTCA",
    "AGCA", "CGCA", "TGCA", "GGCA",
    "AATA", "CATA", "TATA", "GATA",
    "ACTA", "CCTA", "TTCA", "GTCA",
    "ATTA", "CTTA", "TTTA", "GTTA",
45     "AGTA", "CGTA", "TGTA", "GGTA",
    "AAGA", "CAGA", "TAGA", "GAGA",
    "ACGA", "CCGA", "TCGA", "GCGA",
    "ATGA", "CTGA", "TTGA", "GTGA",
    "AGGA", "CGGA", "TGGA", "GGGA",
50     "AAAC", "CAAC", "TAAC", "GAAC",
    "ACAC", "CCAC", "TCAC", "GCAC",
    "ATAC", "CTAC", "TTAC", "GTAC",
    "AGAC", "CGAC", "TGAC", "GGAC",
    "AACC", "CACC", "TACC", "GGCC",
55     "ACCC", "CCCC", "TCCC", "GGCC",
    "ATCC", "CTCC", "TTCC", "GTCC",
    "AGCC", "CGCC", "TGCC", "GGCC",
    "AATC", "CATC", "TATC", "GATC",
    "ACTC", "CCTC", "TTCT", "GTCT",
60     "ATTC", "CTTC", "TTTC", "GTTC",
    "AGTC", "CGTC", "TGTC", "GGTC",
    "AAGC", "CAGC", "TAGC", "GAGC",
    "ACGC", "CCGC", "TCGC", "GCGC",
    "ATGC", "CTGC", "TTGC", "GTGC",
65     "AGGC", "CGGC", "TGGC", "GGGC",
    "AAAT", "CAAT", "TAAT", "GAAT",
    "ACAT", "CCAT", "TCAT", "GCAT",
    "ATAT", "CTAT", "TTAT", "GTAT",
    "AGAT", "CGAT", "TGAT", "GGAT",
70     "AACT", "CACT", "TACT", "GACT",
    "ACCT", "CCCT", "TCCT", "GCCT",
    "ATCT", "CTCT", "TTCT", "GTCT",
    };
}

```

Oct 24, 05 12:36

RepeatCount.java

Page 2/4

```

75      "AGCT", "CGCT", "TGCT", "GGCT",
      "AATT", "CATT", "TATT", "GATT",
      "ACTT", "CCTT", "TCTT", "GCTT",
      "ATTT", "CTTT", "TTTT", "GTTT",
      "AGTT", "CGTT", "TGTT", "GGTT",
80      "AAGT", "CAGT", "TAGT", "GAGT",
      "ACGT", "CCGT", "TCGT", "GCGT",
      "ATGT", "CTGT", "TTGT", "GTGT",
      "AGGT", "CGGT", "TGGT", "GGGT",
      "AAAG", "CAAG", "TAAG", "GAAG",
85      "ACAG", "CCAG", "TCAG", "GCAG",
      "ATAG", "CTAG", "TTAG", "GTAG",
      "AGAG", "CGAG", "TGAG", "GGAG",
      "AACG", "CACG", "TACG", "GACG",
      "ACCG", "CCCG", "TCCG", "GCCG",
90      "ATCG", "CTCG", "TTCG", "GTCC",
      "AGCG", "CGCG", "TGCG", "GGCG",
      "AATG", "CATG", "TATG", "GATG",
      "ACTG", "CCTG", "TCTG", "GCTG",
      "ATTG", "CTTG", "TTTG", "GTTG",
95      "AGTG", "CGTG", "TGTG", "GGTG",
      "AAGG", "CAGG", "TAGG", "GAGG",
      "ACGG", "CCGG", "TCGG", "GCGG",
      "ATGG", "CTGG", "TTGG", "GTGG",
      "AGGG", "CGGG", "TGGG", "GGGG",
100     "AAAG", "CAAG", "TAAG", "GAAG",
      "ACAG", "CCAG", "TCAG", "GCAG",
      "ATAG", "CTAG", "TTAG", "GTAG",
      "AGAG", "CGAG", "TGAG", "GGAG",
      "AACG", "CACG", "TACG", "GACG",
105     "ACCG", "CCCG", "TCCG", "GCCG",
      "ATCG", "CTCG", "TTCG", "GTCC",
      "AGCG", "CGCG", "TGCG", "GGCG",
      "AATG", "CATG", "TATG", "GATG",
      "ACTG", "CCTG", "TCTG", "GCTG",
110     "ATTG", "CTTG", "TTTG", "GTTG",
      "AGTG", "CGTG", "TGTG", "GGTG",
      "AAGG", "CAGG", "TAGG", "GAGG",
      "ACGG", "CCGG", "TCGG", "GCGG",
      "ATGG", "CTGG", "TTGG", "GTGG",
115     "AGGG", "CGGG", "TGGG", "GGGG",
    };

    public void getRepeats(String refdna, int orf, int[] threshold) {
        refdna = refdna.substring(orf-1,refdna.length());

120        String dna;

        System.out.println("Begin Processing. Repeat Length 1, ORF " + orf + ":");
        for (int k=0; k<repeat1Array.length; k++) {
125            dna = refdna;
            int counter = 0;

            while (dna.length() != 0 && dna.indexOf(repeat1Array[k])
130            != -1) {
                dna = dna.substring(dna.indexOf(repeat1Array[k])
, dna.length());
                int tcounter = 0;
                while (dna.length() != 0 && repeat1Array[k].equal
135            s(dna.substring(0,1)) {
                    tcounter++;
                    dna = dna.substring(1,dna.length());
                }
                if (tcounter > threshold[0]) {
                    counter++;
                }
140            if (counter != 0) {
                System.out.println("Found\" + counter + "\" repeats
for repeat sequence\" + repeat1Array[k] + "\"");
            }
        }
    }

```

Oct 24, 05 12:36

RepeatCount.java

Page 3/4

```

        System.out.println();
        System.out.println("Begin Processing. Repeat Length 2, ORF " + orf + ":");
145        for (int k=0; k<repeat2Array.length; k++) {
            dna = refdna;
            int counter = 0;

            while (dna.length() > 1 && dna.indexOf(repeat2Array[k])
150            != -1) {
                dna = dna.substring(dna.indexOf(repeat2Array[k])
, dna.length());
                int tcounter = 0;
                while (dna.length() > 1 && repeat2Array[k].equal
155            s(dna.substring(0,2)) {
                    tcounter++;
                    dna = dna.substring(2,dna.length());
                }
                if (tcounter > threshold[1]) {
                    counter++;
                }
160            if (counter != 0) {
                System.out.println("Found\" + counter + "\" repeats
for repeat sequence\" + repeat2Array[k] + "\"");
            }
        }
165        System.out.println();
        System.out.println("Begin Processing. Repeat Length 3, ORF " + orf + ":");
        for (int k=0; k<repeat3Array.length; k++) {
            dna = refdna;
            int counter = 0;

            while (dna.length() > 2 && dna.indexOf(repeat3Array[k])
170            != -1) {
                dna = dna.substring(dna.indexOf(repeat3Array[k])
, dna.length());
                int tcounter = 0;
                while (dna.length() > 2 && repeat3Array[k].equal
175            s(dna.substring(0,3)) {
                    tcounter++;
                    dna = dna.substring(3,dna.length());
                }
                if (tcounter > threshold[2]) {
                    counter++;
                }
180            if (counter != 0) {
                System.out.println("Found\" + counter + "\" repeats
for repeat sequence\" + repeat3Array[k] + "\"");
            }
        }
185        System.out.println();
        System.out.println("Begin Processing. Repeat Length 4, ORF " + orf + ":");
        for (int k=0; k<repeat4Array.length; k++) {
            dna = refdna;
            int counter = 0;

            while (dna.length() > 3 && dna.indexOf(repeat4Array[k])
190            != -1) {
                dna = dna.substring(dna.indexOf(repeat4Array[k])
, dna.length());
                int tcounter = 0;
                while (dna.length() > 3 && repeat4Array[k].equal
195            s(dna.substring(0,4)) {
                    tcounter++;
                    dna = dna.substring(4,dna.length());
                }
                if (tcounter > threshold[3]) {
                    counter++;
                }
200            if (counter != 0) {
                System.out.println("Found\" + counter + "\" repeats
for repeat sequence\" + repeat4Array[k] + "\"");
            }
        }
    }

```

Oct 24, 05 12:36

RepeatCount.java

Page 4/4

```
205         if (counter != 0) {
                System.out.println("Found\" + counter + "\" repeats
for repeat sequence\" + repeat4Array[k] + "\");
            }
        }
210     }
}
}
```