



Project
on
DNA sequence Checking Algorithm

Submitted By

Student's name	Student ID
Fahim Arefin	2020-1-60-052
Sadia Rahman Ani	2020-2-60-172
Samiha Masrura	2020-1-60-066
Maslinia Sujnus	2019-3-60-013

Department of Computer Science and Engineering
Course Title: Algorithm
Course Code: CSE246
Section: 01

Submitted To

Dr. Taskeed Jabid
Chairperson & Associate Professor
Department of Computer Science and Engineering
East West University

Submission Date

September 18, 2022

Contents

Topic		Page
Abstract		3
Introduction	• Rabin-Karp algorithm	3-5
	• How Rabin-Karp Algorithm Works?	
	• Rabin-Karp algorithm working process	
Project discursion	• code	6-12
	• code Text	
	• output	
Conclusion		13

❖ Abstract

Annie is the mother she lost her child in the park then she complained to the authority but authority had found that there Is also six same age other girls there. So now what Annie should do to find her daughter? DNA sequence test is the only option left. We can use Rabin-karp algorithm to match the DNA sequence of Annie and her daughter.

❖ Introduction

- **Rabin-Karp algorithm:**

Rabin-Karp algorithm is an algorithm used for searching/matching patterns in the text using a hash function. Unlike Naive string match algorithm, it does not travel through every character in the initial phase rather it filters the characters that do not match and then performs the comparison.

- **How Rabin-Karp Algorithm Works?**

A sequence of characters is taken and checked for the possibility of the presence of the required string. If the possibility is found then, character matching is performed.

- **Rabin-Karp algorithm working process:**

Step1: Text

A	B	C	C	D	D	A	E	F	G
---	---	---	---	---	---	---	---	---	---

Pattern:

C	D	D
---	---	---

Step2: Let us assign a numeric value for the characters we will be using in the problem. Here, we have taken first ten alphabets only (i.e. A to J).

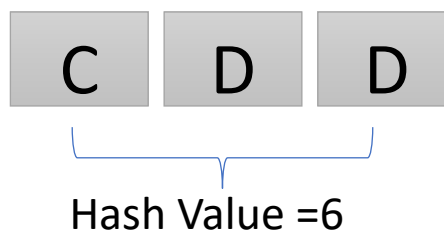
A	B	C	D	E	F	G	H	I	J
1	2	3	4	5	6	7	8	9	10

Step3:

n be the length of the pattern and m be the length of the text. Here, $m=10$ and $n=3$
Let d be the number of characters in the input set. Here, we have taken input set {A, B, C, ..., J}. So, $d=10$ You can assume any suitable value for d.

Step4:

Calculate the hash value of the pattern



$$\begin{aligned}
 \text{hash value for pattern}(p) &= \sum(v * d^{m-1}) \bmod 13 \\
 &= ((3 * 10^2) + (4 * 10^1) + (4 * 10^0)) \bmod 13 \\
 &= 344 \bmod 13 \\
 &= 6
 \end{aligned}$$

Step5:

Calculate the hash value for the text-window of size m

For the first window ABC,

$$\begin{aligned}
 \text{hash value for text}(t) &= \sum(v * d^{n-1}) \bmod 13 \\
 &= ((1 * 10^2) + (2 * 10^1) + (3 * 10^0)) \bmod 13 \\
 &= 123 \bmod 13 \\
 &= 6
 \end{aligned}$$

Step6:

Compare the hash value of the pattern with the hash value of the text. If they match then, character-matching is performed. In the above

examples, the hash value of the first window (i.e. t) matches with p so, go for character matching between ABC and CDD. Since they do not match so, go for the next window.

Step6:

We calculate the hash value of the next window by subtracting the first term and adding the next term as shown below.

$$\begin{aligned} t &= ((1 * 10^2) + ((2 * 10^1) + (3 * 10^0)) * 10 + (3 * 10^0)) \bmod 13 \\ &= 233 \bmod 13 \\ &= 12 \end{aligned}$$

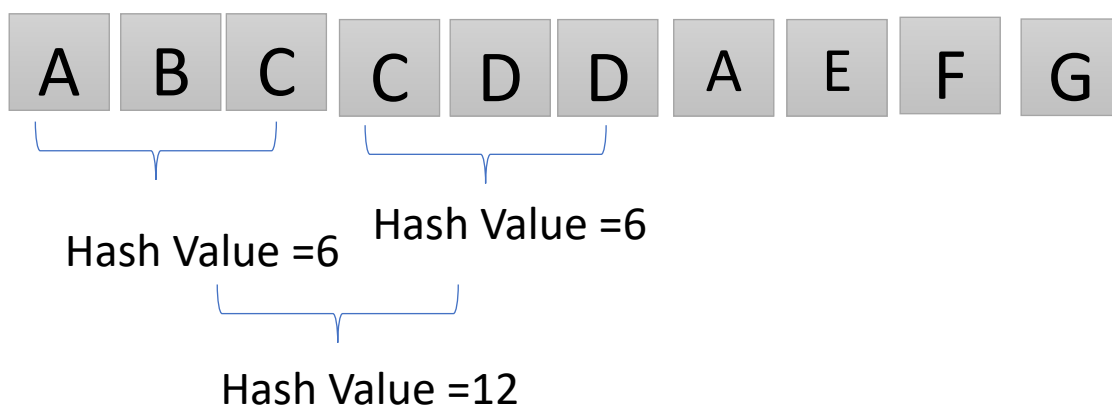
In order to optimize this process, we make use of the previous hash value in the following way.

$$\begin{aligned} t &= ((d * (t - v [\text{character to be removed}] * h) + v [\text{character to be added}]) \bmod 13 \\ &= ((10 * (6 - 1 * 9) + 3) \bmod 13 \\ &= 12 \end{aligned}$$

Where, $h = d^{m-1} = 10^{3-1} = 100$.

For BCC, $t = 12 (\neq 6)$. Therefore, go for the next window.

After a few searches, we will get the match for the window CDA in the text.



❖ Project discursion:

At first 50 sequence of Annie's DNA will take as input from file for check:

```
DNA SEQUENCE OF ANNIE:
-----
BCVJJJXGUJKQJJIUOLWSXXESZUWYSERVJRJVXHSKTZFFSEQIRLP[
-----
Number of DNA sequences of ANNIE:50
```

Now our program will take input from will take input of that six children's DNA sequence accordingly and checked them if the sequence matched with Annie then it will show the Index place from where she can found her baby, if the sequence doesn't match it will show the output of not matching.

- Codes:

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  #define d 50
4  #define dl 4
5  int q=13;
6  char *text=new char [d];
7  char *pattern=new char[dl];
8
9  class ProjectOneEight
10 {
11 public:
12     void myfileWriteforText ();
13     void myfileReadforText ();
14     void project ();
15
16
17 };
18 void ProjectOneEight::myfileWriteforText ()
19 {
20     ofstream myfile("C:\\Users\\Fahim Arefin\\Desktop\\projectCse246\\fahims.txt");
21     srand(time(NULL));
22
23
24
25     for(int i=0; i<d; i++)
26     {
27
28         int hold=1+(rand()%26);
29         char a=65+hold;
30         myfile<<a<<endl;
31
32     }
33
34     myfile.close();
35 }
```

```

37
38 void ProjectOneEight::myfileReadforText()
39 {
40
41     char data;
42
43     ifstream myReadFile("C:\\Users\\Fahim Arefin\\Desktop\\projectCse246\\fahims.txt");
44     cout<<"DNA SEQUENCE OF ANNIE:"<<endl;
45
46     cout<<"-----"<<endl;
47     int count=0;
48     for(int i=0; i<d; i++)
49     {
50         myReadFile>>data;
51         cout<<data;
52         text[i]=data;
53         count++;
54     }
55     cout<<endl;
56     cout<<"-----"<<endl;
57     cout<<"Number of DNA sequences of ANNIE:"<<count;
58     cout<<endl;
59 }
60 void ProjectOneEight::project()
61 {
62
63     int m = 4 ;
64     int n = 50 ;
65     int p = 0;
66     int t = 0;
67     int h = 1;
68
69

```

```

70     cout<<"Enter the child DNA SEQUENCE:"<<endl;
71     for(int i=0; i<m; i++)
72     {
73         cin>>pattern[i];
74     }
75
76
77
78     int i,j;
79     for (i = 0; i < m - 1; i++)
80         h = (h * d) % q;
81
82
83
84     for (i = 0; i < m; i++)
85     {
86         p = (d * p + pattern[i]) % q;
87         t = (d * t + text[i]) % q;
88     }
89
90
91     for (i = 0; i <= n - m; i++)
92     {
93
94
95         if ( p == t )
96         {
97
98             for (j = 0; j < m; j++)
99             {
100                 if (text[i+j] != pattern[j])
101                 {
102                     break;
103                 }
104

```

```

106         }
107
108
109         if (j == m)
110         {
111             cout<<"ANNIE GOT HER BABY AT INDEX PLACE: "<< i<<endl;
112             break;
113         }
114     }
115
116 }
117
118
119 if ( i < n-m )
120 {
121     t = (d*(t - text[i]*h) + text[i+m])%q;
122
123     if (t < 0)
124         t = (t + q);
125 }
126 else
127 {
128     cout<<"Not matching"<<endl;
129 }
130 }
131 }
132
133
134 }
135
136
137 int main()
138 {
139
140     ProjectOneEight obj1;
141     obj1.myfileWriteforText();
142     obj1.myfileReadforText();
143     cout<<"_____ "<<endl;
144     cout<<"There is six girls in the park so you have to"<<endl;
145     cout<<"give input DNA sequence of those girls to find "<<endl;
146     cout<<"Annie's daughter"<<endl;
147     cout<<"_____ "<<endl;
148     for(int i=0; i<6; i++)
149     {
150         obj1.project();
151     }
152 }
153
154

```

- **Codes text:**

```
#include<bits/stdc++.h>
```

```
using namespace std;
```

```
#define d 50
```

```
#define d1 4
```



```

int q=13;

char *text=new char [d];
char *pattern=new char[d1];

class ProjectOneEight
{
public:
    void myfileWriteforText();
    void myfileReadforText();
    void project();
};

void ProjectOneEight::myfileWriteforText()
{
    ofstream myfile("C:\\Users\\Fahim Arefin\\Desktop\\projectCse246\\fahims.txt");
    srand(time(NULL));
    for(int i=0; i<d; i++)
    {
        int hold=1+(rand()%26);
        char a=65+hold;
        myfile<<a<<endl;
    }
    myfile.close();
}

void ProjectOneEight::myfileReadforText()
{
    char data;
    ifstream myReadFile("C:\\Users\\Fahim Arefin\\Desktop\\projectCse246\\fahims.txt");
    cout<<"DNA SEQUENCE OF ANNIE:"<<endl;

    cout<<"-----"<<endl;
}

```

```

int count=0;
for(int i=0; i<d; i++)
{
    myReadFile>>data;
    cout<<data;
    text[i]=data;
    count++;
}
cout<<endl;
cout<<"-----"<<endl;
cout<<"Number of DNA sequences of ANNIE:"<<count;
cout<<endl;
}
void ProjectOneEight::project()
{
    int m = 4 ;
    int n = 50 ;
    int p = 0;
    int t = 0;
    int h = 1;
    cout<<"Enter the child DNA SEQUENCE:"<<endl;
    for(int i=0; i<m; i++)
    {
        cin>>pattern[i];
    }
    int i,j;
    for (i = 0; i < m - 1; i++)
        h = (h * d) % q;
    for (i = 0; i < m; i++)

```

```

{
    p = (d * p + pattern[i]) % q;
    t = (d * t + text[i]) % q;
}
for (i = 0; i <= n - m; i++)
{
    if ( p == t )
    {
        for (j = 0; j < m; j++)
        {
            if (text[i+j] != pattern[j])
            {
                break;
            }
        }
        if (j == m)
        {
            cout<<"ANNIE GOT HER BABY AT INDEX PLACE: "<< i<<endl;
            break;
        }
    }
    if ( i < n-m )
{
    t = (d*(t - text[i]*h) + text[i+m])%q;

    if (t < 0)
        t = (t + q);
    }
else

```

```

    {
        cout<<"Not matching"<<endl;
    }
}

int main()
{
    ProjectOneEight obj1;
    obj1.myfileWriteforText();
    obj1.myfileReadforText();
    cout<<"_____ "<<endl;
    cout<<"There is six girls in the park so you have to"<<endl;
    cout<<"give input DNA sequence of those girls to find "<<endl;
    cout<<"Annie's daughter"<<endl;
    cout<<"_____ "<<endl;
    for(int i=0; i<6; i++)
    { obj1.project();
    }
}

```

- **Output:**

In this way we by checking the DNA sequence with Annie,
Annie will able to get child.

```
ZNKQSQRJZHTRCOBNGFVIOISOWYINDEPYMSNILCCWZHUTUCCWR
-----
Number of DNA sequences of ANNIE:50

There is six girls in the park so you have to
give input DNA sequence of those girls to find
Annie's daughter

Enter the child DNA SEQUENCE:
FJGJ
Not matching
Enter the child DNA SEQUENCE:
AGCC
Not matching
Enter the child DNA SEQUENCE:
AGTI
Not matching
Enter the child DNA SEQUENCE:
AGUU
Not matching
Enter the child DNA SEQUENCE:
GFVI
ANNIE GOT HER BABY AT INDEX PLACE: 17
Enter the child DNA SEQUENCE:
AGUI
Not matching

Process returned 0 (0x0)    execution time : 49.686 s
Press any key to continue.
```

- ❖ **Conclusion:** After finishing this project it clear to us that how our project can perform as an DNA sequence checking algorithm.