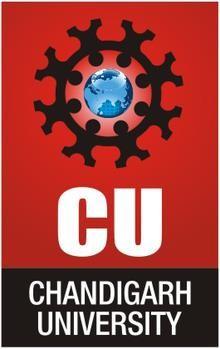
**CHANDIGARH UNIVERSITY**

UNIVERSITY INSTITUTE OF ENGINEERING

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**



|  |  |
| --- | --- |
| **Submitted By:                                                                          Submitted To:**  Yash Gupta ER. Monika(E12802) | |
| **Subject Name** | Design Analysis and Algorithm |
| **Subject Code** | 20CSP\_312 |
| **Branch** | CSE |
| **Semester** | 5th |

**LAB -INDEX**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Sr.No** | **Program** | **Date** | **Evaluation** | | | | **Sign** |
| **LW(12)** | **VV(8)** | **FW(10)** | **Total (30)** |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

**Experiment 3.3**

**1. Aim/Overview of the practical:**

Code and Analyze to find all occurrences of a pattern P in a given string S.

1. **Task to be done/which logistics used:**

In this problem we will solve the pattern problem using rabin-karp algorithm .

1. **Algorithm/Flowchart:**

n = t.length m = p.length h = dm-1 mod q p = 0 t0 = 0 for i = 1 to m p = (dp + p[i]) mod q t0 = (dt0 + t[i]) mod q for s = 0 to n - m if p = ts if p[1.....m] = t[s + 1..... s + m] print "pattern found at position" s If s < n-m ts + 1 = (d (ts - t[s + 1]h) + t[s + m + 1]) mod q

**4. Steps for experiment/practical/Code:**

#include <string.h>

#include <iostream> using namespace std;

#define d 10

void rabinKarp(char pattern[], char text[], int q) { int m = strlen(pattern); int n = strlen(text); int i, j; int p = 0; int t = 0; int h = 1;

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 << "Pattern is found at position: " << i + 1 << endl;

}

if (i < n - m) { t = (d \* (t - text[i] \* h) + text[i + m]) % q;

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

}

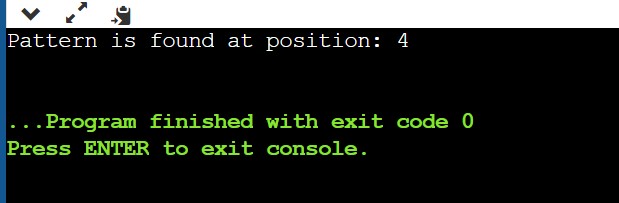
}

}

int main() { char text[] = "ABCCDDAEFG"; char pattern[] = "CDD"; int q = 13; rabinKarp(pattern, text, q);

}

**5. Result/Output/Writing Summary:**



6. **Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No. | Parameters | Marks Obtained | Maximum Marks |
| 1. |  |  |  |
| 2. |  |  |  |
| 3. |  |  |  |