**EXPT 8b**: PATTERN SEARCHING

**Program**:

#include<stdio.h>

#include<string.h>

#define NO\_OF\_CHARS 256

int getNextState(char \*pat, int M, int state, int x)

{

if (state < M && x == pat[state])

return state+1;

int ns, i;

for (ns = state; ns > 0; ns--)

{

if (pat[ns-1] == x)

{

for (i = 0; i < ns-1; i++)

if (pat[i] != pat[state-ns+1+i])

break;

if (i == ns-1)

return ns;

}

}

return 0;

}

void computeTF(char \*pat, int M, int TF[][NO\_OF\_CHARS])

{

int state, x;

for (state = 0; state <= M; ++state)

for (x = 0; x < NO\_OF\_CHARS; ++x)

TF[state][x] = getNextState(pat, M, state, x);

}

void search(char \*pat, char \*txt)

{

int M = strlen(pat);

int N = strlen(txt);

int TF[M+1][NO\_OF\_CHARS];

computeTF(pat, M, TF);

int i, state=0;

for (i = 0; i < N; i++)

{

state = TF[state][txt[i]];

if (state == M)

printf ("\n Pattern found at index %d", i-M+1);

}

}

int main()

{

char \*txt = "AABAACAADAABAAABAA";

char \*pat = "AABA";

search(pat, txt);

return 0;

}

**Output:**

