Working of NAÏVE PATTERN MATCHING

int nfind(char \*string, char \*pat)

{

// match the last character of the pattern first and then match from the beginning

int i,j,start=0;

int lasts=strlen(string)-1;

lastp=strlen(pat)-1;

int endmatch=lastp;

for(i=0;endmatch<=lasts;endmatch++,start++)

{

if string[endmatch]==pat[lastp])

for (j=0,i=start;j<lastp &string[i]==pat[j];i++,j++)

;

if(j==lastp)

return start;

}

return-1;

}

Example---1-----------PATTERN FOUND

string

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| A | B | C | A | B | A | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 |

pat

|  |  |  |  |
| --- | --- | --- | --- |
| C | A | B | \0 |
| 0 | 1 | 2 | 3 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| i | j | start | lasts | lastp | endmatch |  |  |
|  |  | 0 | 5 | 2 | 2 |  |  |
| 0 |  |  |  |  |  | for(i=0;endmatch<=lasts;endmatch++,start++) | 2<=5 |
|  |  |  |  |  |  | if string[endmatch]==pat[lastp]) | string[2]==pat[2]  C==B |
|  |  |  |  |  |  | if(j==lastp) | garbagevalue==2 |
|  |  | 1 |  |  | 3 | for(i=0;endmatch<=lasts;endmatch++,start++) | 3<=5 |
|  |  |  |  |  |  | if string[endmatch]==pat[lastp]) | string[3]==pat[2]  A==B |
|  |  |  |  |  |  | if(j==lastp) | garbagevalue==2 |
|  |  | 2 |  |  | 4 | for(i=0;endmatch<=lasts;endmatch++,start++) | 4<=5 |
|  |  |  |  |  |  | if string[endmatch]==pat[lastp]) | string[4]==pat[2]  B==B |
| 2 | 0 |  |  |  |  | for (j=0,i=start;j<lastp &&string[i]==pat[j];i++,j++) | 0<2&&string[2]==pat[0]  C==C |
| 3 | 1 |  |  |  |  |  | 1<2 && string[3]==pat[1]  A==A |
| 4 | 2 |  |  |  |  |  | 2<2 && string[4]==pat[2]  CONDITION FAILED |
|  |  |  |  |  |  | if(j==lastp) | 2==2  return start----2 |
|  |  | 3 |  |  | 5 | for(i=0;endmatch<=lasts;endmatch++,start++) | 5<=5 |
|  |  |  |  |  |  | if string[endmatch]==pat[lastp]) | string[5]==pat[2]  A==B |
|  |  |  |  |  |  | if(j==lastp) | 2==2  return start----2 |
|  |  | 4 |  |  | 6 | for(i=0;endmatch<=lasts;endmatch++,start++) | 6<=5  CONDITION FAILED |

Example---2-----------PATTERN NOT FOUND

string

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| A | B | C | A | B | A | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 |

pat

|  |  |  |  |
| --- | --- | --- | --- |
| X | A | B | \0 |
| 0 | 1 | 2 | 3 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| i | j | start | lasts | lastp | endmatch |  |  |
|  |  | 0 | 5 | 2 | 2 |  |  |
| 0 |  |  |  |  |  | for(i=0;endmatch<=lasts;endmatch++,start++) | 2<=5 |
|  |  |  |  |  |  | if string[endmatch]==pat[lastp]) | string[2]==pat[2]  C==B |
|  |  |  |  |  |  | if(j==lastp) | garbagevalue==2 |
|  |  | 1 |  |  | 3 | for(i=0;endmatch<=lasts;endmatch++,start++) | 3<=5 |
|  |  |  |  |  |  | if string[endmatch]==pat[lastp]) | string[3]==pat[2]  A==B |
|  |  |  |  |  |  | if(j==lastp) | garbagevalue==2 |
|  |  | 2 |  |  | 4 | for(i=0;endmatch<=lasts;endmatch++,start++) | 4<=5 |
|  |  |  |  |  |  | if string[endmatch]==pat[lastp]) | string[4]==pat[2]  B==B |
| 2 | 0 |  |  |  |  | for (j=0,i=start;j<lastp &&string[i]==pat[j];i++,j++) | 0<2&&string[2]==pat[0]  C==X |
|  |  |  |  |  |  | if(j==lastp) | 0==2 |
|  |  | 3 |  |  | 5 | for(i=0;endmatch<=lasts;endmatch++,start++) | 5<=5 |
|  |  |  |  |  |  | if string[endmatch]==pat[lastp]) | string[5]==pat[2]  A==B |
|  |  |  |  |  |  | if(j==lastp) | 0==2 |
|  |  | 4 |  |  | 6 | for(i=0;endmatch<=lasts;endmatch++,start++) | 6<=5 |
|  |  |  |  |  |  |  | return ---1  not found |

Example---3------------unique characters in txt and pat----txt is not revisited

txt

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | E | F | G | H | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |

pat

|  |  |  |  |
| --- | --- | --- | --- |
| D | E | F | \0 |
| 0 | 1 | 2 | 3 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| i | j | start | lasts | lastp | endmatch |  |  |
|  |  | 0 | 7 | 2 | 2 |  |  |
| 0 |  |  |  |  |  | for(i=0;endmatch<=lasts;endmatch++,start++) | 2<=7 |
|  |  |  |  |  |  | if string[endmatch]==pat[lastp]) | string[2]==pat[2]  C==F |
|  |  |  |  |  |  | if(j==lastp) | garbagevalue==2 |
|  |  | 1 |  |  | 3 | for(i=0;endmatch<=lasts;endmatch++,start++) | 3<=7 |
|  |  |  |  |  |  | if string[endmatch]==pat[lastp]) | string[3]==pat[2]  D==F |
|  |  |  |  |  |  | if(j==lastp) | garbagevalue==2 |
|  |  | 2 |  |  | 4 | for(i=0;endmatch<=lasts;endmatch++,start++) | 4<=7 |
|  |  |  |  |  |  | if string[endmatch]==pat[lastp]) | string[4]==pat[2]  E==F |
|  |  |  |  |  |  | if(j==lastp) | garbagevalue==2 |
|  |  | 3 |  |  | 5 | for(i=0;endmatch<=lasts;endmatch++,start++) | 5<=7 |
|  |  |  |  |  |  | if string[endmatch]==pat[lastp]) | string[5]==pat[2]  F==F |
| 3 | 0 |  |  |  |  | for (j=0,i=start;j<lastp &&string[i]==pat[j];i++,j++) | 0<2&&string[3]==pat[0]  D==D |
| 4 | 1 |  |  |  |  |  | 0<2&&string{4]==pat[1]  E==E |
| 5 | 2 |  |  |  |  |  | 2<2 &&stringt[5]==pat[2]  CONDITION FAILED |
|  |  |  |  |  |  | if(j==lastp) | 2==2  RETURN START ---return 3 |
|  |  |  |  |  |  |  | 0==2 |
|  |  | 3 |  |  | 5 | for(i=0;endmatch<=lasts;endmatch++,start++) | 5<=5 |
|  |  |  |  |  |  | if string[endmatch]==pat[lastp]) | string[5]==pat[2]  A==B |
|  |  |  |  |  |  | if(j==lastp) | 0==2 |
|  |  | 4 |  |  | 6 | for(i=0;endmatch<=lasts;endmatch++,start++) | 6<=5 |
|  |  |  |  |  |  |  | return ---1  not found |

SPECIAL CASE ------------duplicate characters in txt ----txt is revisited

txt

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

txt

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

txt

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

txt

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

txt

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |  | 10 | 11 | 12 |

Pat NO MATCH

|  |
| --- |
| 9 |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

txt REVISTING / BACKTRACKING IN TXT from index 4 to 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat NO MATCH

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

txt NO MATCH

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

txt NO MATCH

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

txt

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

txt

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

txt

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

txt NO MATCH

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

txt REVISTING / BACKTRACKING IN TXT from index 7 to 5

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

NO MATCH

txt

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

txt

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

txt

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

txt

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

txt

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

txt

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | A | B | C | A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

pat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | F | \0 |
| 0 | 1 | 2 | 3 | 4 | 5 |

void fail(char \*pat)

{   // compute the pattern’s failure function

int n=strlen(pat);

failure[0]=-1;

for (j=1; j <n;j++)

i=failure[j-1];

while((pat[j]!=pat[i+1]) && i>==0)

i=failure[i];

if (pat[j]==pat[i+1])

failure[j]=i+1;

else

failure[j]=-1;

}

}

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Pat | A | A | A | A | B | A | A | C | D | \0 |
| Failure[] | -1 |  |  |  |  |  |  |  |  |  |

N=9

Iteration-------1

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Pat | A | A | A | A | B | A | A | C | D | \0 |
| for (j=1; j <n;j++) j=1;j<9;  i=failure[j-1]----i=failure[0]---------i=-1 | | | | | | | | | | |
| while((pat[j]!=pat[i+1]) && i>==0)  i=failure[i];  -------While fails as i=-1 | | | | | | | | | | |
| if (pat[j]==pat[i+1])---pat[1]==pat[0]----A==A  failure[j]=i+1-----failure[1]=0 |  |  |  |  |  |  |  |  |  |  |
| Failure[] | -1 | 0 |  |  |  |  |  |  |  |  |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

Iteration-------2

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Pat | A | A | A | A | B | A | A | C | D | \0 |
| for (j=1; j <n;j++) j=2;j<9;  i=failure[j-1]----i=failure[1]---------i=0 | | | | | | | | | | |
| while((pat[j]!=pat[i+1]) && i>==0)  i=failure[i];  -------While fails as pat[2]!=pat[1]----A!=A | | | | | | | | | | |
| if (pat[j]==pat[i+1])---pat[2]==pat[1]----A==A  failure[j]=i+1-----failure[2]=1 |  |  |  |  |  |  |  |  |  |  |
| Failure[] | -1 | 0 | 1 |  |  |  |  |  |  |  |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

Iteration-------3

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Pat | A | A | A | A | B | A | A | C | D | \0 |
| for (j=1; j <n;j++) j=3;j<9;  i=failure[j-1]----i=failure[2]---------i=1 | | | | | | | | | | |
| while((pat[j]!=pat[i+1]) && i>==0)  i=failure[i];  -------While fails as pat[3]!=pat[2]----A!=A | | | | | | | | | | |
| if (pat[j]==pat[i+1])---pat[3]==pat[2]----A==A  failure[j]=i+1-----failure[3]=2 |  |  |  |  |  |  |  |  |  |  |
| Failure[] | -1 | 0 | 1 | 2 |  |  |  |  |  |  |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

Iteration-------4

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Pat | A | A | A | A | B | A | A | C | D | \0 |
| for (j=1; j <n;j++) j=4;j<9;  i=failure[j-1]----i=failure[3]---------i=2 | | | | | | | | | | |
| while((pat[j]!=pat[i+1]) && i>==0)  i=failure[i];  ------- pat[4]!=pat[3]----B!=A && 2>=0  I=failure[2]-----1  Pat[4]!=pat[2]--------B!=A && 1>=0  I=failure[1]------0  Pat[4]!=pat[1]---------B!=A &&0>=0  I=failure[0]------1  -------While fails as i=-1 | | | | | | | | | | |
| if (pat[j]==pat[i+1])---pat[4]==pat[0]----B==A  else  failure[j]=-1-------failure[4]=-1 |  |  |  |  |  |  |  |  |  |  |
| Failure[] | -1 | 0 | 1 | 2 | -1 |  |  |  |  |  |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

Iteration-------5

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Pat | A | A | A | A | B | A | A | C | D | \0 |
| for (j=1; j <n;j++) j=5;j<9;  i=failure[j-1]----i=failure[4]---------i=-1 | | | | | | | | | | |
| while((pat[j]!=pat[i+1]) && i>==0)  i=failure[i];  -------While fails as i=-1 | | | | | | | | | | |
| if (pat[j]==pat[i+1])---pat[5]==pat[0]----A==A  failure[j]=i+1-----failure[5]=0 |  |  |  |  |  |  |  |  |  |  |
| Failure[] | -1 | 0 | 1 | 2 | -1 | 0 |  |  |  |  |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

Iteration-------6

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Pat | A | A | A | A | B | A | A | C | D | \0 |
| for (j=1; j <n;j++) j=6;j<9;  i=failure[j-1]----i=failure[5]---------i=0 | | | | | | | | | | |
| while((pat[j]!=pat[i+1]) && i>==0)  i=failure[i];  pat[6]!=pat[1]-------A!=A  -------While fails as A!=A | | | | | | | | | | |
| if (pat[j]==pat[i+1])---pat[6]==pat[1]----A==A  failure[j]=i+1-----failure[6]=1 |  |  |  |  |  |  |  |  |  |  |
| Failure[] | -1 | 0 | 1 | 2 | -1 | 0 | 1 |  |  |  |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

Iteration-------7

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Pat | A | A | A | A | B | A | A | C | D | \0 |
| for (j=1; j <n;j++) j=7;j<9;  i=failure[j-1]----i=failure[6]---------i=1 | | | | | | | | | | |
| while((pat[j]!=pat[i+1]) && i>==0)  i=failure[i];  pat[7]!=pat[2]-------C!=A  i=failure[1]--------i=0  while((pat[j]!=pat[i+1]) && i>==0)  i=failure[i];  pat[7]!=pat[1]-------C!=A  i=failure[0]---------i=-1  -------While fails as i>=0 | | | | | | | | | | |
| if (pat[j]==pat[i+1])---pat[7]==pat[0]----C==A  else part  failure[j]=-1-------failure[7]=-1 |  |  |  |  |  |  |  |  |  |  |
| Failure[] | -1 | 0 | 1 | 2 | -1 | 0 | 1 | -1 |  |  |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

Iteration-------8

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Pat | A | A | A | A | B | A | A | C | D | \0 |
| for (j=1; j <n;j++) j=8;j<9;  i=failure[j-1]----i=failure[7]---------i=-1 | | | | | | | | | | |
| while((pat[j]!=pat[i+1]) && i>==0)  i=failure[i];  -------While fails as i>=0 | | | | | | | | | | |
| if (pat[j]==pat[i+1])---pat[8]==pat[0]----D==A  else part  failure[j]=-1-------failure[8]=-1 |  |  |  |  |  |  |  |  |  |  |
| Failure[] | -1 | 0 | 1 | 2 | -1 | 0 | 1 | -1 | -1 |  |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

Iteration-------8

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Pat | A | A | A | A | B | A | A | C | D | \0 |
| Failure[] | -1 | 0 | 1 | 2 | -1 | 0 | 1 | -1 | -1 |  |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

int pmatch(char \*string, char \* pat)

{ //Kunth,Morris, pratt algorithm

int i=0,j=0;

int lens=strlen(string);

int lenp=strlen(pat);

while(i<lens  && j< lenp) {

if string[i]==pat[j]){

i++;j++;}

else if (j==0) i++;

else j=failure[j-1]+1;

}

return ((j==lenp) ?(i-lenp):-1);

}

pat

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | 0 | 1 | 2 | 3 |
| Pat | A | A | B | \0 |
| Failure[] | -1 | 0 | -1 |  |

txt

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Txt | A | B | A | B | B | A | A | B | A | A | \0 |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  | I | J | Lens | Lenp |
|  | 0 | 0 | 10 | 3 |
| While i<lens && j<lenp ---0<10 && 0<3 |  |  |  |  |
| If string[i]==pat[j]----string[0]==pat[0]----A==A |  |  |  |  |
|  | 1 | 1 |  |  |
| While i<lens && j<lenp ---1<10 && 1<3 |  |  |  |  |
| If string[i]==pat[j]---- string[1]==pat[1]----B==A |  |  |  |  |
| Else part----  J=failure[j-1]+1----j=failure[0]+1—j=0 | 1 | 0 |  |  |
| While i<lens && j<lenp ---1<10 && 0<3 |  |  |  |  |
| If string[i]==pat[j]---- string[1]==pat[0]----B==A |  |  |  |  |
| J==0, I++ | 2 |  |  |  |
| While i<lens && j<lenp ---2<10 && 0<3 |  |  |  |  |
| If string[i]==pat[j]---- string[2]==pat[0]----A==A | 3 | 1 |  |  |
| While i<lens && j<lenp ---3<10 && 1<3 |  |  |  |  |
| If string[i]==pat[j]---- string[3]==pat[1]----B==A |  |  |  |  |
| Else part  J=fail;ure[j-1]+1----j=failure[0]+1----j=0 |  | 0 |  |  |
| While i<lens && j<lenp ---3<10 && 0<3 |  |  |  |  |
| If string[i]==pat[j]---- string[3]==pat[0]----B==A |  |  |  |  |
| Else if---j==0;i++ | 4 |  |  |  |
| While i<lens && j<lenp ---4<10 &&0<3 |  |  |  |  |
| If string[i]==pat[j]---- string[4]==pat[0]----B==A |  |  |  |  |
| Else if---j==0;i++ | 5 |  |  |  |
| While i<lens && j<lenp ---5<10 && 0<3 |  |  |  |  |
| If string[i]==pat[j]---- string[5]==pat[0]----A==A | 6 | 1 |  |  |
| While i<lens && j<lenp ---6<10 && 1<3 |  |  |  |  |
| If string[i]==pat[j]---- string[6]==pat[1]----A==A | 7 | 2 |  |  |
| While i<lens && j<lenp ---7<10 &&2<3 |  |  |  |  |
| If string[i]==pat[j]---- string[7]==pat[2]----A==A | 8 | 3 |  |  |
| While i<lens && j<lenp ---5<10 && 3<3------fails |  |  |  |  |

Return (j==lenp)-----i-lenp----8-3---5-----------index of the pattern