DAA Practical No 4

Aim: Write C/C++ code for following algorithm with explanation:

1)Travelling salesman Problem

2)BF String matching Algorithm

3)Exhaustive Search Algorithm

1)Travelling salesman Problem

-- >Program Code:

Go Run Terminal Help • tsp.cpp - DAA - Visual Studio Code

tsp.cpp

p4_impalgoritms > tsp.cpp > TSP(int [][V], int)

```cpp
//Travelling Salesman Problem
//A traveller needs to visit all the cities from a list,
//where distances between all the cities are known &each city should be visited just once.
//What is the shortest possible route that he visits each city exactly once and
//also he returns to the origin city?

#include <bits/stdc++.h>
using namespace std;
#define V 4
int TSP (int graph[][V], int s)
{
    vector<int> vertex;
    for (int i = 0; i < V; i++)
        if (i != s)
            vertex.push_back(i);

    int min_path = INT_MAX;
    do {
        int current_pathweight = 0;
        int k = s;
        for (int i = 0; i < vertex.size(); i++) {
            current_pathweight += graph[k][vertex[i]];
            k = vertex[i];
        }
        current_pathweight += graph[k][s];
        // update minimum
        min_path = min(min_path, current_pathweight);

    } while (
        next_permutation(vertex.begin(), vertex.end()));
    return min_path;
}
```

```
int main()
{
    int graph[][V] = { { 0, 3, 4, 5 },
                       { 3, 0, 6, 7 },
                       { 4, 6, 0, 8 },
                       { 5, 7, 8, 0 } };
    int s = 0;
    cout << "Minimum disatance required : "<<TSP (graph, s) << endl;
    return 0;
}
```

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

PS C:\Users\DELL\Desktop\DAA> cd "c:\Users\DELL\Desktop\DAA\p4_impalgoritms"
PS C:\Users\DELL\Desktop\DAA\p4_impalgoritms> & .\"tsp.exe"
Minimum disatance required : 22
PS C:\Users\DELL\Desktop\DAA\p4_impalgoritms>
```

2)BF String matching Algorithm

-- > Program Code:

Go Run Terminal Help string_match.c - DAA - Visual Studio Code

string_match.c

p4_impalgoritms > string_match.c > string_search(char *, char *, int, int)

```c
#include <stdio.h>
#include <string.h>

// Brute-force string search function
void string_search(char *string, char *pattern, int s_len, int p_len) {
    int i = 0;
    int found = 0;
    for(; i < s_len - p_len + 1; i++) {
        int j = 0;
        for (; j < p_len; j++) {
            if (string[i + j] != pattern[j])
                break;
        }
        if (j == p_len) {   // If we have reached end of pattern, we have found the pattern in string
            found = 1;
            break;
        }
    }
    if (found)
        printf("Found pattern at index: %d\n", i);
    else
        printf("Could not find the pattern\n");
}
```

```
24
25  // Driver function
26  int main() {
27      char *string = "ABCABAB ABABABAABAC";
28      char *pattern = "ABABAABA";
29      int s_len = strlen(string);
30      int p_len = strlen(pattern);
31
32      string_search(string, pattern, s_len, p_len);
33
34      return 0;
35  }
```

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

PS C:\Users\DELL\Desktop\DAA> cd "c:\Users\DELL\Desktop\DAA\p4_impalgoritms"
PS C:\Users\DELL\Desktop\DAA\p4_impalgoritms> & .\"string_match.exe"
Found pattern at index: 10
PS C:\Users\DELL\Desktop\DAA\p4_impalgoritms>
```

3)Exhaustive Search Algorithm

-- > Program Code:

Go Run Terminal Help • exhaustiveSearch.cpp - DAA - Visual Studio Code

exhaustiveSearch.cpp

p4_impalgoritms > exhaustiveSearch.cpp > ...

```
1   //calculation of the number of chickens and the number of rabbits
2   //The exhaustive method is used to solve the problem
3   #include<iostream>
4   using namespace std;
5
6   int getAns(int head,int foot,int *chicken,int * rabbit)
7   {
8       int re,i,j;
9       re=0;
10      for(i=0;i<=head;i++){
11          j=head-i;
12          if(i*2+j*4==foot){
13          re=1;   //To find the answer
14          *chicken=i;
15          *rabbit=j;
16          }
17      }
18      return re;
19  }
```

```
int main()
{
    int chicken,rabbit,head,foot;
    int re;
    cout<<" Please enter the number of heads: ";
    cin>>head;
    cout<<" Please enter the number of feet: ";
    cin>>foot;
    re=getAns(head,foot,&chicken,&rabbit);
    if(re==1){
        cout<<" Total number of chickens present are "<<chicken<<
                    " and total number of rabbits are "<<rabbit<<endl;
    }
    else{
        cout<<" Unsolvable! "<<endl;
    }
    return 0;
}
```

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
PS C:\Users\DELL\Desktop\DAA> cd "c:\Users\DELL\Desktop\DAA\p4_impalgoritms"
PS C:\Users\DELL\Desktop\DAA\p4_impalgoritms> & .\"exhaustiveSearch.exe"
 Please enter the number of heads: 10
 Please enter the number of feet: 24
 Total number of chickens present are 8 and total number of rabbits are 2
PS C:\Users\DELL\Desktop\DAA\p4_impalgoritms> & .\"exhaustiveSearch.exe"
 Please enter the number of heads: 20
 Please enter the number of feet: 10
 Unsolvable!
PS C:\Users\DELL\Desktop\DAA\p4_impalgoritms>
```