Experiment 4: (searching and sorting)

**Q1) Write a program to implement linear search on array elements using UDF**

**#include<stdio.h>**

**int linear\_search(int\*, int, int);**

**main()**

**{**

**int array[100], search, c, n, position;**

**printf("Enter the number of elements in array\n");**

**scanf("%d",&n);**

**printf("Enter %d numbers\n", n);**

**for ( c = 0 ; c < n ; c++ )**

**scanf("%d",&array[c]);**

**printf("Enter the number to search\n");**

**scanf("%d",&search);**

**position = linear\_search(array, n, search);**

**if ( position == -1 )**

**printf("%d is not present in array.\n", search);**

**else**

**printf("%d is present at location %d.\n", search, position+1);**

**return 0;**

**}**

**int linear\_search(int \*pointer, int n, int find)**

**{**

**int c;**

**for ( c = 0 ; c < n ; c++ )**

**{**

**if ( \*(pointer+c) == find )**

**return c;**

**}**

**return -1;**

**}**

**Q2) Write a program to implement binary search on array elements using UDF**

**#include<stdio.h>**

**#include<conio.h>**

**#define MAX\_SIZE 5**

**void binary\_search(int[],int);**

**int main()**

**{**

**int arr\_search[MAX\_SIZE], i,element;**

**printf("Simple Binary Search Example - Array and Functions\n");**

**printf("\nEnter %d Elements for Searching : \n", MAX\_SIZE);**

**for (i = 0; i < MAX\_SIZE; i++)**

**scanf("%d", &arr\_search[i]);**

**printf("Enter Element to Search : ");**

**scanf("%d", &element);**

**binary\_search(arr\_search,element);**

**getch();**

**}**

**void binary\_search(int fn\_arr[],int element)**

**{**

**int f = 0, r = MAX\_SIZE,mid;**

**while (f <= r)**

**{**

**mid = (f+r)/2;**

**if (fn\_arr[mid] == element)**

**{**

**printf("\nSearch Element : %d : Found : Position : %d.\n", element, mid+1);**

**break;**

**}**

**else if (fn\_arr[mid] < element)**

**f = mid + 1;**

**else**

**r = mid - 1;**

**}**

**if (f > r)**

**printf("\nSearch Element : %d : Not Found \n", element);**

**}**

**Sample Output:**

**Simple Binary Search Example - Array and Functions**

**Enter 5 Elements for Searching :**

**1001**

**1020**

**3002**

**4001**

**5000**

**Enter Element to Search : 3002**

**Search Element : 3002 : Found : Position : 3.**

**Q3) write a program to implement selection sort on a given list of array elements.**

**#include <stdio.h>**

**void selection(int arr[], int n)**

**{**

**int i, j, small;**

**for (i = 0; i < n-1; i++)    // One by one move boundary of unsorted subarray**

**{**

**small = i; //minimum element in unsorted array**

**for (j = i+1; j < n; j++)**

**if (arr[j] < arr[small])**

**small = j;**

**// Swap the minimum element with the first element**

**int temp = arr[small];**

**arr[small] = arr[i];**

**arr[i] = temp;**

**}**

**}**

**void printArr(int a[], int n) /\* function to print the array \*/**

**{**

**int i;**

**for (i = 0; i < n; i++)**

**printf("%d ", a[i]);**

**}**

**int main()**

**{**

**int a[] = { 12, 31, 25, 8, 32, 17 };**

**int n = sizeof(a) / sizeof(a[0]);**

**printf("Before sorting array elements are - \n");**

**printArr(a, n);**

**selection(a, n);**

**printf("\nAfter sorting array elements are - \n");**

**printArr(a, n);**

**return 0;**

**}**

**Q4) Write a program to input a string and sort the alphabets in ascending order.**

**#include <stdio.h>**

**#include <string.h>**

**int main ()**

**{**

**char string[100];**

**printf("\n\t Enter the string : ");**

**scanf("%s",string);**

**char temp;**

**int i, j;**

**int n = strlen(string);**

**for (i = 0; i < n-1; i++)**

**{**

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

**{**

**if (string[i] > string[j])**

**{**

**temp = string[i];**

**string[i] = string[j];**

**string[j] = temp;**

**}**

**}**

**}**

**printf("The sorted string is : %s", string);**

**return 0;**

**}**