|  |  |  |
| --- | --- | --- |
| **Name: Naren Chandrashekhar** | **SRN: PES2UG20CS216** | **Section: G** |
| **Date: 8/07/2021** | **Week Number: 10** |

|  |  |
| --- | --- |
| **1** | Implement Binary Search using call back when there is more than one constraint to check for.  a) Search for a number if the number is even  b) Search for a number if the number is less than 22.  **Input:**  enter the element to be searched  18  **Output:**  It is even and found at 2 position  It is less than 22 and found at 2 position  **Input:**  enter the element to be searched  56  **Output:**  It is even and found at 8 position  not found  **Input:**  enter the element to be searched  53  **Output:**  not found  not found |
|  | **Program:**  /\*#include<stdio.h>  int search(int[],int,int,int,int(\*p)(int));  int iseven(int);  int islessthan22(int);  int main()  {  int a[]={20,30,40,45,55,89,101};  int n;  int key;  int pos;  n=sizeof(a)/sizeof(\*a);  printf("Enter the element to be searched:");  scanf("%d",&key);  pos=search(a,0,n-1,key,iseven);  if(pos==-1)  printf("Element not found\n");  else  printf("Element is even and found at %d\n",pos);  pos=search(a,0,n-1,key,islessthan22);  if(pos==-1)  printf("Element not found\n");  else  printf("Element is less than 22 and found at %d\n",pos);  return 0;  }  int iseven(int x)  {  return x%2==0;  }  int islessthan22(int x)  {  return x<22;  }  int search(int a[],int low,int high,int key,int(\*p)(int))  {  int pos=-1;  int mid;  if(low>high)  return pos;  else  mid=(low+high)/2;  if(a[mid]==key && p(key))  pos=mid;  else if(a[mid]>key)  return search(a,low,mid-1,key,p);  else  return search(a,mid+1,high,key,p);  return pos;  }\*/  #include <stdio.h>  int binsearch(int \*a,int low,int high,int key,int (\*fn)(int));  int even(int x);  int num(int x);  int main()  {  int a[]={11,13,18,19,22,33,55,66,77,88};  int key;  int n=sizeof(a)/sizeof(\*a);  printf("Enter the element to be searched: ");  scanf("%d",&key);  int pos=binsearch(a,0,n-1,key,even);  if(pos==-1)  printf("Number is not even\n");  else  printf("It is even and found at %d position\n",pos);  pos=binsearch(a,0,n-1,key,num);  if(pos==-1)  printf("Number is not less than 22\n");  else  printf("It is less than 22 and found at %d position\n",pos);  return 0;  }  int even(int x)  {  return x%2==0;  }  int num(int x)  {  return x<22;  }  int binsearch(int \*a,int low,int high,int key,int (\*fn)(int))  {  int mid; int res=-1; int m;  if(low>high)  return res;  else  {  mid=(low+high)/2;  if(a[mid]==key && fn(key))  return mid;  else if(a[mid]>key)  return binsearch(a,low,mid-1,key,fn);  else  return binsearch(a,mid+1,high,key,fn);  }  } |
|  | **Output Screenshot:** |
| 2 | Write a program to copy the contents of one file to another using command line arguments  (Instruction to be given in the command line)  >a abc.txt def.txt  (abc.txt is the file having contents which will be copied to the file def.txt) |
|  | **Program:**  #include <stdio.h>  #include <stdlib.h>  int main()  {  FILE \*fptr1, \*fptr2;  char filename[100], c;  printf("Enter the filename to open for reading \n");  scanf("%s", filename);  fptr1 = fopen(filename, "r");  if (fptr1 == NULL)  {  printf("Cannot open file %s \n", filename);  exit(0);  }  printf("Enter the filename to open for writing \n");  scanf("%s", filename);  fptr2 = fopen(filename, "w");  if (fptr2 == NULL)  {  printf("Cannot open file %s \n", filename);  exit(0);  }  c = fgetc(fptr1);  while (c != EOF)  {  fputc(c, fptr2);  c = fgetc(fptr1);  }  printf("\nContents copied to %s", filename);  fclose(fptr1);  fclose(fptr2);  return 0;  } |
|  | **Output Screenshot:** |
| 3 | Write a program using enumerated types which when given today's date will print out tomorrow's date.  **Input:**  Enter a date (number 3 letter lower case month e.g. 31 jan)  30 nov  **Output:**  Tomorrow is 1 dec  **Input:**  Enter a date (number 3 letter lower case month e.g. 31 jan)  31 dec  **Output:**  Tomorrow is 1 jan |
|  | **Program:**  #include <stdio.h>  #include <string.h>  #include <stdlib.h>  enum months{jan=1,feb,mar,apr,may,jun,jul,aug,sep,oct,nov,dec}  month;  static char  \*month\_out[]={"NOT\_MONTH","jan","feb","mar","apr","may","jun","jul","aug","sep","oct","nov","dec"};  static int days\_in\_month[]={-1,31,28,31,30,31,30,31,31,30,31,30,31};  enum months translate(char\*);  int check(int,enum months);  void tomorrow(int,enum months);  int main()  {  int day;  char mon[4];  printf("Enter a date(number followed by month in 3 letter lowercase e.g.31 jan) \n");  scanf("%d %s",&day,mon);  month=translate(mon);  if(!check(day,month))  tomorrow(day,month);  return(0);  }  enum months translate(char\*m)  {  if(strcmp(m,"jan")==0)  return jan;  else if(strcmp(m,"feb")==0)  return feb;  else if(strcmp(m,"mar")==0)  return mar;  else if(strcmp(m,"apr")==0)  return apr;  else if(strcmp(m,"may")==0)  return may;  else if(strcmp(m,"jun")==0)  return jun;  else if(strcmp(m,"jul")==0)  return jul;  else if(strcmp(m,"aug")==0)  return aug;  else if(strcmp(m,"sep")==0)  return sep;  else if(strcmp(m,"oct")==0)  return oct;  else if(strcmp(m,"nov")==0)  return nov;  else if(strcmp(m,"dec")==0)  return dec;  }  int check(int day,enum months month\_in)  {  if((day<1)||(day>days\_in\_month[month]))  {  printf("Error:Invalid Input %d %s \n",day,month\_out[month\_in]);  return 1;  }  else  return 0;  }  void tomorrow(int day,enum months month\_in)  {  if(day < days\_in\_month[month\_in])  printf("Tomorrow is %d %s \n",day+1,month\_out[month\_in]);  else if((day==days\_in\_month[month\_in])&&(month\_out[month\_in]!="dec"))  printf("Tomorrowis 1 %s \n",month\_out[month\_in+1]);  else  printf("Tomorrow is 1 %s \n","jan");  } |
|  | **Output Screenshot:** |