WACP to find a element in an array along with the execution time of the program.

```
#include <stdio.h>
#include <time.h>
int main()
{
   float time_in_sec;
   int array[100], search, c, n;
   clock_t start, end;

   start = clock();
   printf("Enter number of elements in array\n");
   scanf("%d", &n);

   printf("Enter %d integer(s)\n", n);

   for (c = 0; c < n; c++)
      scanf("%d", &array[c]);

   printf("Enter a number to search\n");
   scanf("%d", &search);

   for (c = 0; c < n; c++)
   {
      if (array[c] == search)    /* If required element is found */
      {
         printf("%d is present at location %d.\n", search, c+1);
         break;
      }
   }
   if (c == n)
      printf("%d isn't present in the array.\n", search);
   end = clock();
   time_in_sec = (float)(end-start)/CLOCKS_PER_SEC;
   printf("The execution time is %f",time_in_sec);
   return 0;
}
```

WACP to generate random numbers of n elements and print it in an array.

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
   int n,I,J,Duplicate,Numbers[1000];
   FILE *fptr;
   printf("Enter how many random numbers you want to generate");
   scanf("%d",&n);
   //for loop to generate a complete set of n random numbers
   for (I = 0; I < n; I++)
   {
      Duplicate = 0; // set check to false

      // do while loop used to generate random numbers until a distinct random number is
generated
      do
      {
         Numbers[I] = (rand()%n) + 1; // generates a random number 1 - n and stores it into
Numbers[I]

         // for loop used to check the other numbers in set for any repeats
         for (J = I - 1; J > -1; J--) // works backwards from the recently generated element to
element 0
            if (Numbers[I] == Numbers[J]) //checks if number is already used
               Duplicate = 1; //sets Duplicate to true to indicate there is a repeat

      } while (Duplicate); //loops until a new, distinct number is generated
   }
   for(I=0;I<n;I++)
      printf("%d ",Numbers[I]);
   fptr = (fopen("random_numbers.txt", "w"));
   if(fptr == NULL)
   {
      printf("Error!");
      exit(1);
   }
   for(I=0;I<n;I++)
```

```
      fprintf(fptr,"%d ", Numbers[I]);

   fclose(fptr);
}
```

WACP to read and store numbers from files into an array.

```
#include <stdio.h>

int main (void)
{
   int nums[1000];   //up to 1000 element int array
   FILE *fp1;      //file pointer
   int i,n;

   printf("Enter how many elements you want to read");
   scanf("%d",&n);

   //****************** code logic starts here ***************
   for(i=0;i<n;i++)   //initialize array elements with 0
      nums[i]=0;
   i=0;       //re-initialize the array index at i=0
   if ((fp1=fopen("random_numbers.txt","r"))==NULL)    //Checks if the file exists
   {
      printf("random_numbers.txt failed to open\n");
      return 1;
   }
   else
   {
      while((fscanf(fp1,"%d",&nums[i]))!=EOF)     //scanf, check and continue untill EOF
         i++;
      for(i=0;i<n;i++)
      {
         printf("nums[%d] is %d\n",i,nums[i]);
      }
   }
   return 0;
 }
```