Shiv Nadar University

CSD101: Introduction to Computing and Programming

Quiz #3

Max marks: 30 13-Nov-2021
Time:12.00-12.30pm + 5mins (grace period for submission)

1. *The answers should be written on BB itself where space is provided for an answer.* **No uploads**.
2. *You* **must submit by 12.35pm**. *Late submissions may not be graded.*
3. *You can submit* **at most twice**. *The second submission will be graded if you have submitted twice.*
4. *If you are having problems on BB while doing/submitting the quiz refreshing the screen often helps.*

1. The code given on the next page implements an efficient way ($O(log_2(n))$ to search for a target value in an already sorted array (in ascending order). It returns the index of the target value in the array if it is present in the array and -1 otherwise.

   For example, if the array is: $[2, 5, 9, 11, 23, 42]$ and the target value is 23 then the output will be 4. If we search for 7 the output will be -1.

   In each round the program compares the target element with the middle element in the current array and returns the index if it matches. Otherwise, it continues the search either in the first half or second half of the current array. It eliminates half the current array in each round.

   In the `main()` function the program reads the size of the array, the elements in the array (to be input in increasing order) and the target value to be searched.

   There are some missing parts in the program indicated by `?n?` for $1 \leq n \leq 10$. Fill in those parts so that the program works correctly.

   [Hint: pay attention to comments in the code.]

   `You don't have to reproduce the program just write the part number and your corresponding answer for that part number.`

   Note that there is no partial marking for any part. If your answer to a part is correct you get 3 marks, if wrong 0 marks.

   [3×10=30]

```
#include <stdio.h>
#include <stdlib.h>

int search(int *arrp, int size, int val) {
/*Searches for element val in the integer array pointed to by
  arrp. The array is sorted in increasing/ ascending order.
  Returns the index of the element if val is present in the
  array else returns -1.
*/
   int *strtp=arrp, ?1?;
   //strtp - starting address
   while (size> ?2?) {
      mp=?3?;//address of mid point of current array
      if (val==?3? ) return ?4?;//target value found
      else
         if (val>?5?) {//search 2nd half
            strtp+=(size-1)/2+1;
            size=?6?;
         }
         else //search 1st half
            size=?7?;
   }
   return ?8?;
}
int main() {
   int size,v;
   printf("Give size of 1D array (+ve integer)=");
   scanf("%d",&size);
   if (size<1) {printf("Illegal size"); exit(0);}
   int a[size];
   printf("Give array elements in ascending order = ");
   for (int i=0; i<size; i++) scanf("%d",?9?);
   printf("Give search element = ");
   scanf("%d",&v);
   printf("%d\n",search(?10?,size,v));
   exit(0);
}
```

Solution:

There were 11 blanks instead of 10. ?3? was repeated twice. This is written as 3a. and 3b. So, total marks are 33 instead of 30.

```
1. *mp
2.  0
3a. strtp+(size-1)/2
3b. *mp
4. (int)(mp-arrp)
5. *mp
6. size-size/2
7. (size-1)/2
8. -1
9. &a[i]
10, &a[0]
```