

1D Array Suggestive Common Problems and few solutions

1. Write a program in C to find the sum of all elements of the array.

```
#include <stdio.h>

int main()
{
    int index, sum=0, size;
    printf("Enter the size of the array: ");
    scanf("%d", &size);
    int arr[size];

    for(index = 0; index<size; index++)
    {
        scanf("%d",&arr[index]);
        sum += arr[index];
    }

    printf("sum of all elements in the array: %d\n", sum);
    return 0;
}
```

2. Write a program in C to find the minimum/maximum value of the array. Output the position where it is found also.

```
//storing the values of the array simply
for(index = 0; index<size; index++)
{
    scanf("%d",&arr[index]);
}

//now finding out the minimum of the array
//this loop below could also be combined with the above
one
int min = arr[0], position;
for(index=0; index < size; index++)
{
    if(arr[index] < min)
    {
        min = arr[index];
        position = index;
    }
}
```

```
printf("minimum value of the array: %d\nFound at  
position: %d\n", min, position);
```

3. Write a program in C to print only the unique elements in an array

```
//storing values in the array
for(index = 0; index<size; index++)
{
    scanf("%d",&arr[index]);
}

for(index=0; index<size; index++)
{
    counter=0;
    for(int j=0,k=size; j<k+1; j++)
    {
        //Increment the counter when a duplicate is found
        if (index!=j)
        {
            if(arr[index]==arr[j])
            {
                counter++;
            }
        }
        if(counter==0)
        //coming out of inner loop, without finding any duplicates
        {
            printf("%d ",arr[index]);
        }
    }
}
```

4. Write a program in C to find the second largest element in an array.

5. Write a program in C to separate odd and even integers in separate arrays.

```
int even_index, odd_index, odd[size], even[size];
even_index = odd_index = 0;
for(index=0; index<size; index++)
{
    if (arr[index]%2 == 0)
    {
        even[even_index] = arr[index];
        even_index++;
    }
    else
    {

```

```

        odd[odd_index] = arr[odd_index];
        odd_index++;
    }

    // the variables even_index and odd_index now carries
the size of those smaller arrays
    printf("\nThe Even elements are : \n");
    for(index=0; index<even_index; index++)
    {
        printf("%d ", even[index]);
    }
    printf("\nThe Odd elements are : \n");
    for(index=0; index<odd_index; index++)
    {
        printf("%d ", odd[index]);
    }

```

6. Write a program in C to sort elements of array in ascending/descending order.
7. Write a program in C to delete an element at desired position from an array

```

printf("\nInput the position where to delete: ");
scanf("%d", &position);

/*---- locate the position of i in the array -----*/
index=0;
while(index!=position-1) index++;

/*---- the position of i in the array will be replaced
by the
        value of its right */
while(index<size)
{
    arr[index]=arr[index+1];
    index++;
}-
size--;
printf("\nThe new list is : ");
for(index=0; index<size; index++)
{
    printf("  %d", arr[index]);
}

```

8. Write a program in C to insert a New value in the array.

```

printf("Input the value to be inserted : ");
scanf("%d", &value);

```

```

    printf("Input the Position, where the value to be
inserted :");
    scanf("%d",&position);

    for(index=size; index>=position; index--)
    {
        arr[index]= arr[index-1];
    }
    /* insert value at given position */
    arr[position-1] = value;

    printf("\n\nAfter Insert the element the new list is
:\n");
    for(index=0; index<=size; index++)
        printf("%d ",arr[index]);

```

9. Write a program in C to find the common values between two separate arrays.

10. Write a program in C to find the first repeated element in an array

```

int index_repeat = -1;
//check first repeated element
for(index=0; index<size; index++)
{
    for(j=index+1; j<size; j++)
    {
        if(arr[index]==arr[j])
        {
            value = arr[j];
            index_repeat = j;
            break;
        }
    }

    if(index_repeat != -1)
        break;
}
if(index_repeat != -1)
    printf("Value: %d is repeated @ index: %d\n",
value, index_repeat);
else
    printf("There is no repeated element\n");

```

PLEASE PRACTICE MORE VARIATIONS OF PROBLEMS FROM THE INTERNET ALSO.

-----GOOD LUCK-----