🡪Easy 🡪Medium 🡪Hard

**Practical 1** (Topic: Pointers & Structures)

**Class-work:**

* Write a program to create and store records of students dynamically (using pointers) and display them.

Solution:

*# include <string.h>   
# include <stdio.h>   
struct student   
{   
        char name[10];   
        int m[3];   
        int total;   
        char result[5];   
}\*p,\*s;   
        void main()   
    {   
              int i,j,l,n;   
              clrscr();   
              printf("Enter the no. of students : ");   
              scanf("%d",&n);   
              p=(struct student\*)malloc(n\*sizeof(struct student));   
              s=p;   
              for(i=0;i<n;i++)   
                  {   
                             printf("Enter a name : ");   
                             scanf("%s",&p->name);   
                             p-> total=0;l=0;   
                             for(j=0;j<3;j++)   
                                {   
                                     one:printf("Enter Marks of %d Subject : ",j+1);   
                                     scanf("%d",&p->m[j]);   
                                     if((p->m[j])>100)   
                                        {   
                                            printf("Wrong Value Entered");   
                                            goto one;   
                                        }   
                                            p->total+=p->m[j];   
                                            if(p->m[j]<40)   
                                                l=1;   
                                  }   
                                            if(l==0)   
                                              strcpy(p->result,"PASS");   
                                            else   
                                              strcpy(p->result,"FAIL");   
                                              p++;   
                      }   
                                              for(i=0;i<n;i++)   
                                                  {   
                                                        printf("\n%s\t%s",s->name,s->result);   
                                                        s++;   
                                                  }   
                                                        getch();   
       }*

[](http://ecomputernotes.com/images/Structure-of-N-Students-using-Dynamic-Memory-Allocation.jpg)

Solution without dynamically:

#include <stdio.h>

struct student

{

char name[50];

int roll;

float marks;

} s;

int main()

{

printf("Enter information:\n");

printf("Enter name: ");

scanf("%s", s.name);

printf("Enter roll number: ");

scanf("%d", &s.roll);

printf("Enter marks: ");

scanf("%f", &s.marks);

printf("Displaying Information:\n");

printf("Name: ");

puts(s.name);

printf("Roll number: %d\n",s.roll);

printf("Marks: %.1f\n", s.marks);

return 0;

}

Enter information:

Enter name: Jack

Enter roll number: 23

Enter marks: 34.5

Displaying Information:

Name: Jack

Roll number: 23

Marks: 34.5

**Home-work:**

1. Write a program to ask the user for 10 integer numbers and display the even ones.
2. Write a program to evaluate student’s performance using structure; where structure contains name and total\_marks.
3. Write a program to calculate sum and average of all element of 2-D array.
4. Write a program to ask the user for 10 real numbers and display the average of the positive ones and the average of the negative ones
5. Write a program to interchange two variables using pointers, using the concept of invoking functions by passing the pointers.
6. Write a Program to create a structure for Employee containing appropriate fields for ID, Salary and Name. Prompt the user to enter the details of “n” employees and instantiate the structure and store the references in an array of pointers. Now write a function to display the details of an employee with given ID.
7. WAP to delete duplicate elements from an array of “n” integers.
8. WAP that returns the sum of the elements in a specific row of an array. The array, number of filled-in columns and which row is to be totaled should be passed as parameters.
9. Write a program to print names and number of employees who have 5 year or more experience and salary less than Rs. 10,000/- using an array of structures.
10. Write a program to find frequency of each element in an array using pointer.

\* C program to count frequency of each element of array

1. \*/
2. #**include** <stdio.h>
3. **int** main()
4. {
5. **int** arr[100], freq[100];
6. **int** size, i, j, count;
7. /\* Input size of array \*/
8. printf("Enter size of array: ");
9. scanf("%d", &size);
10. /\* Input elements in array \*/
11. printf("Enter elements in array: ");
12. **for**(i=0; i<size; i++)
13. {
14. scanf("%d", &arr[i]);
15. /\* Initially initialize frequencies to -1 \*/
16. freq[i] = -1;
17. }
18. **for**(i=0; i<size; i++)
19. {
20. count = 1;
21. **for**(j=i+1; j<size; j++)
22. {
23. /\* If duplicate element is found \*/
24. **if**(arr[i]==arr[j])
25. {
26. count++;
27. /\* Make sure not to count frequency of same element again \*/
28. freq[j] = 0;
29. }
30. }
31. /\* If frequency of current element is not counted \*/
32. **if**(freq[i] != 0)
33. {
34. freq[i] = count;
35. }
36. }
37. /\*
38. \* Print frequency of each element
39. \*/
40. printf("\nFrequency of all elements of array : \n");
41. **for**(i=0; i<size; i++)
42. {
43. **if**(freq[i] != 0)
44. {
45. printf("%d occurs %d times\n", arr[i], freq[i]);
46. }
47. }
48. **return** 0;
49. }