

## QUESTION 1

```
#include <stdio.h>

int main()
{
    int arr1[100], arr2[100], mergeArray[100 * 2];
    int size1, size2, mergeSize;
    int index1, index2, mergeIndex;
    int i;

    /* Input size of first array */
    printf("Enter the size of first array : ");
    scanf("%d", &size1);

    /* Input elements in first array */
    printf("Enter elements in first array : ");
    for(i=0; i<size1; i++)
    {
        scanf("%d", &arr1[i]);
    }

    /* Input size of second array */
    printf("\nEnter the size of second array : ");
    scanf("%d", &size2);

    printf("Enter elements in second array : ");
    for(i=0; i<size2; i++)
    {
        scanf("%d", &arr2[i]);
    }

    mergeSize = size1 + size2;

    index1 = 0;
    index2 = 0;
    for(mergeIndex=0; mergeIndex < mergeSize; mergeIndex++)
    {
        if(index1 >= size1 || index2 >= size2)
        {
            break;
        }

        if(arr1[index1] < arr2[index2])
        {
            mergeArray[mergeIndex] = arr1[index1];
            index1++;
        }
        else
        {
            mergeArray[mergeIndex] = arr2[index2];
```

```

        index2++;
    }
}

while(index1 < size1)
{
    mergeArray[mergeIndex] = arr1[index1];
    mergeIndex++;
    index1++;
}
while(index2 < size2)
{
    mergeArray[mergeIndex] = arr2[index2];
    mergeIndex++;
    index2++;
}

printf("\nArray merged in ascending order : ");
for(i=0; i<mergeSize; i++)
{
    printf("%d\t", mergeArray[i]);
}

return 0;
}

```

QUESTION 2 :

```

#include <stdio.h>

int main()
{
    int votes[5][4]={192,48,206,37,
                     147,90,312,21,
                     186,12,121,38,
                     114,21,408,39,
                     267,13,382,29};

    char cand[4]={'A','B','C','D'};

    int row_totals[5];
    int col_totals[4];
    int total_votes;
    float percent[4];
    int row,col;

    for(row=0;row<=4;row++)
    {
        row_totals[row]=0;

        for(col=0;col<=3;col++)
        {
            row_totals[row] += votes[row][col];

```

```

        }
    }

    printf(" Precinct    Candidate    Candidate    Candidate\n");
    printf("Candidate    Total\n");
    printf("          A          B          C          D\n");
    printf("Votes\n");
    for(row=0;row<=4;row++)
    {
        printf("%6d",row+1);
        for(col=0;col<=3;col++)
        {
            printf("%12d",votes[row][col]);
        }
        printf("%11d\n",row_totals[row]);
    }

    return 0;

```

QUESTION : 3

```

}

#include <stdio.h>

struct rainfall{
    float thisyr;
    float pastyr;
};

int main()
{
    float totalthis, totalpast, averagethis, averagepast;
    struct rainfall detail[12];

    for(int i=0;i<12;i++)
    {
        printf("Enter The present past rainfall for %d month:",(i+1));
        scanf("%f %f",&detail[i].thisyr,&detail[i].pastyr);
    }
    printf("\n*****DISPLAY THE DETAIL OF RAINFALL*****:\n");
    printf("JAN\tFEB\tMAR\tAPR\tMAY\tJUNE\tJULY\tAUG\tSEPT\tOCT\t\tNOV\tDEC\n");
    for(int i=0;i<12;i++)
    {
        printf("%1.f\t",detail[i].thisyr);
        totalthis=totalthis+detail[i].thisyr;
    }
    printf("\n");
    for(int i=0;i<12;i++)
    {
        printf("%1.f\t",detail[i].pastyr);
        totalpast=totalpast+detail[i].pastyr;
    }
}

```

```

}
averagethis=totalthis/12;
averagepast=totalpast/12;

printf("TOTAL RAINFALL THIS YEAR:%2.f:\n",totalthis);
printf("TOTAL RAINFALL PAST YEAR%2.f:\n",totalpast);
printf("Average monthly rainfall for this year%2.f:\n",averagethis);
printf("Average monthly rainfall for past year%2.f:\n",averagepast);

    return 0;
}

```

#### QUESTION 4

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

```

int main()
{
    float n;
    printf("Enter the value of n :- ");
    scanf("%f",&n);
    if(n<5.0)
    {
        printf("little or no damage");
    }
    else if(n>=5.0 && n<5.5)
    {
        printf("Some damage");
    }
    else if(n>=5.5 && n<6.5)
    {
        printf("Serious damage walls may crack or wall");
    }else if(n>=6.5 && n<7.5)
    {
        printf("disaster houses and building may collapse");
    }
    else if(n>7.5)
    {
        printf("catastrophe most building destroyed");
    }
    return 0;
}

```