```c
void main()
{
    int m1[10][10];
    int i,j,k, m2[10][10], add[10][10], mult[10][10];
    int r1,c1,r2,c2;
    printf("Enter number of rows & columns of 1st matrix MAX 10\n");
    scanf("%d%d",&r1,&c1);
    printf("Enter number of rows & columns of 2nd matrix MAX 10\n");
    scanf("%d%d",&r2,&c2);
    if(r2==c1)
    {
        printf("Enter rows and columns of First matrix \n");
        printf("Row wise\n");
        for(i=0;i<r1;i++)
        {
            for(j=0;j<c1;j++)
                scanf("%d",&m1[i][j]);
        }
        printf("You have entered the first matrix as follows:\n");
        for(i=0;i<r1;i++)
        {
            for(j=0;j<c1;j++)
                printf("%d\t",m1[i][j]);
            printf("\n");
        }
        printf("Enter rows and columns of Second matrix \n");
        printf("Again row wise\n");
        for(i=0;i<r2;i++)
        {
            for(j=0;j<c2;j++)
                scanf("%d",&m2[i][j]);
        }
        printf("You have entered the second matrix as follows:\n");
        for(i=0;i<r2;i++)
        {
            for(j=0;j<c2;j++)
                printf("%d\t",m2[i][j]);
            printf("\n");
        }
        if(r1==r2&&c1==c2)
        {
            printf("Now we add both the above matrix \n");
            printf("The result of the addition is as follows;\n");
            for(i=0;i<r1;i++)
            {
                for(j=0;j<c1;j++)
                {
                    add[i][j]=m1[i][j]+m2[i][j];
                    printf("%d\t",add[i][j]);
                }
                printf("\n");
            }
        }
        else
        {
            printf("Addition cannot be done. rows != columns \n");
        }
        printf("Now we multiply both the above matrix \n");
        printf("The result of the multiplication is as follows:\n");
```

```
        for(i=0;i<r1;i++)
        {
            for(j=0;j<c2;j++)
            {
                mult[i][j]=0;
                for(k=0;k<r1;k++)
                {
                    mult[i][j]+=m1[i][k]*m2[k][j];
                }
                printf("%d\t",mult[i][j]);
            }
            printf("\n");
        }
    }
    else
    {
        printf("Matrix multiplication cannot be done");
    }
}
```

Binary Search – Υλοποίηση με επανάληψη

```
int binaryLoopSearch(int p[], int searchkey,
      int low, int high)
{
      int middle;
      while ( low <= high )
      {
            middle = (low + high ) / 2;

            if (searchkey == p[middle])
                  return middle;
            else if (searchkey < p[middle] )
                  high = middle - 1;
            else
                  low = middle + 1;
      }
      return -1;
}
```