CSE115L – Computing Concepts Lab

Array declaration, input and output:

```
#include<stdio.h>
int main()
{
    int a[5]={2,4,8,3,5};
    printf("%d\n",a[0]);
    printf("%d\n",a[1]);
    printf("%d\n",a[2]);
    printf("%d\n",a[3]);
    return 0;
}
```

```
#include<stdio.h>
int main()
{
    int a[2];
    scanf("%d",&a[0]);
    scanf("%d",&a[1]);
    printf("%d\n",a[0]);
    printf("%d\n",a[1]);
    return 0;
}
```

```
#include<stdio.h>
int main()
{
    float a[2];
    scanf("%f",&a[0]);
    scanf("%f",&a[1]);
    printf("%.2f\n",a[0]);
    printf("%.2f\n",a[1]);
    return 0;
}
```

Accessing array elements using loop:

```
#include<stdio.h>
int main()
{
    int n;
    printf("Enter the length of Array:");
    scanf("%d",&n);
    int a[n];
    for(int i=0;i<n;i++)
    {
        scanf("%d",&a[i]);
    }
    for(int i=0;i<n;i++)
    {
        printf("a[%d]= %d\n",i,a[i]);
    }
    return 0;
}
```

```
#include<stdio.h>
int main()
{
    int n,sum=0;
    printf("Enter the length of Array:");
    scanf("%d",&n);
    int a[n];
    for(int i=0;i<n;i++)
    {
        scanf("%d",&a[i]);
    }
    for(int i=0;i<n;i++)
    {
        sum=sum+a[i];
    }
    printf("The sum is: %d",sum);
    return 0;
}
```

Passing arrays as function arguments:

```
#include<stdio.h>
double getAverage(int arr[], int size);
int main ()
{
    int balance[5] = {1000, 2, 3, 17, 50};
    double avg;

    avg = getAverage( balance, 5 ) ;

    printf( "Average value: %f ", avg );

    return 0;
}
```

```
double getAverage(int arr[], int size)
{
    int    i;
    double avg;
    double sum;

    for (i = 0; i < size; ++i)
    {
        sum += arr[i];
    }

    avg = sum / size;

    return avg;
}
```

Problems:

1. Write a program that takes 5 integers as input, stores them in an array and prints them in the reverse order of the input.

Sample Output:

```
Enter array elements: 2 3 5 4 9
Reversed order: 9 4 5 3 2
```

2. Write a program that takes 5 integers as input, stores them in an array and finds the highest number in the array.

Sample Output:

```
Enter array elements: 2 3 15 4 9
Largest element: 15
```

3. Write a function that searches for value (key) in an array of size defined by the user. The function should print "Found" if the element is in the array and "Not found" otherwise.

```
void search(int arr[], int size, int key);
```

Sample Output 1:

```
Enter array size: 5
Array Elements: 4 3 2 9 12
Search Key: 9
Found
```

Sample Output 2:

```
Enter array size: 5
Array Elements: 4 3 2 9 12
Search Key: 7
Not found
```