DAY – 1:

Reversing a 32 bit signed intergers

```
#include <stdio.h>

#include <limits.h>

int reverse(int n) {

   int reversed = 0;

   while (n != 0) {

      int pop = n % 10;

      n /= 10;

      if (reversed > INT_MAX / 10 || (reversed == INT_MAX / 10 && pop > 7)) return 0;  // Overflow check

      if (reversed < INT_MIN / 10 || (reversed == INT_MIN / 10 && pop < -8)) return 0; // Underflow check

      reversed = reversed * 10 + pop;

   }

   return reversed;

}

int main() {

   int n = 123456789; // Example number

   int result = reverse(n);

   printf("Reversed integer: %d\n", result);

   return 0;

}
```

Check for a valid String

```
#include <stdio.h>

#include <ctype.h>


int isValidString(const char *str) {

   while (*str) {

      if (!isalpha(*str)) return 0; // Return 0 if any non-alphabetic character is found
```

```
        str++;
    }
    return 1; // Return 1 if the entire string is alphabetic
}
int main() {
    char str[] = "HelloWorld"; // Example string
    if (isValidString(str))
        printf("The string is valid.\n");
    else
        printf("The string is not valid.\n”);
    return 0;
}
```

Merging two Arrays

```
#include <stdio.h>
void mergeArrays(int arr1[], int size1, int arr2[], int size2, int merged[]) {
    for (int i = 0; i < size1; i++) merged[i] = arr1[i];
    for (int i = 0; i < size2; i++) merged[size1 + i] = arr2[i];
}
int main() {
    int arr1[] = {1, 2, 3};
    int arr2[] = {4, 5, 6};
    int merged[6];
    mergeArrays(arr1, 3, arr2, 3, merged);
    for (int i = 0; i < 6; i++) printf("%d ", merged[i]);
    return 0;
}
```

Given an array finding duplication values

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

```
void findDuplicates(int arr[], int size) {
    for (int i = 0; i < size - 1; i++) {
        for (int j = i + 1; j < size; j++) {
            if (arr[i] == arr[j]) {
                printf("Duplicate found: %d\n", arr[i]);
                break;
            }
        }
    }
}
int main() {
    int arr[] = {2, 3, 4, 2, 5, 6, 4};
    int size = sizeof(arr) / sizeof(arr[0]);
    findDuplicates(arr, size);
    return 0;
}
```

Merging of list

```
#include <stdio.h>
void merge(int arr1[], int size1, int arr2[], int size2, int merged[]) {
    for (int i = 0; i < size1; i++) merged[i] = arr1[i];
    for (int j = 0; j < size2; j++) merged[size1 + j] = arr2[j];
}
int main() {
    int arr1[] = {1, 3, 5};
    int arr2[] = {2, 4, 6};
    int merged[6];
    merge(arr1, 3, arr2, 3, merged);
    for (int i = 0; i < 6; i++) printf("%d ", merged[i]);
    return 0;
}
```

Given array of reg nos need to search for particular reg no

```c
#include <stdio.h>

int main() {

    int regNos[] = {101, 102, 103, 104, 105};

    int n = sizeof(regNos) / sizeof(regNos[0]);

    int search, found = 0;

    printf("Enter registration number to search: ");

    scanf("%d", &search);

    for (int i = 0; i < n; i++) {

        if (regNos[i] == search) {

            found = 1;

            break;

        }

    }

    if (found)

        printf("Registration number %d found!\n", search);

    else

        printf("Registration number %d not found.\n", search);

    return 0;

}
```

Identify location of element in given array

```c
#include <stdio.h>

int main() {

    int arr[] = {10, 20, 30, 40, 50};

    int n = sizeof(arr) / sizeof(arr[0]);

    int target = 30, index = -1;

    for (int i = 0; i < n; i++) {

        if (arr[i] == target) {

            index = i;

            break;
        }
    }

    if (index != -1)
        printf("Element %d found at index %d.\n", target, index);
    else
        printf("Element %d not found in the array.\n", target);

    return 0;
}
```

Given array print odd and even values

```
#include <stdio.h>
int main() {
    int arr[] = {1, 2, 3, 4, 5, 6};
    int n = sizeof(arr) / sizeof(arr[0]);
    printf("Even numbers: ");
    for (int i = 0; i < n; i++)
        if (arr[i] % 2 == 0) printf("%d ", arr[i]);
    printf("\nOdd numbers: ");
    for (int i = 0; i < n; i++)
        if (arr[i] % 2 != 0) printf("%d ", arr[i]);
    return 0;
}
```

Sum of Fibonacci Series

```
#include <stdio.h>
int main() {
    int n, a = 0, b = 1, sum = 0, temp;
    printf("Enter number of terms: ");
    scanf("%d", &n);
    for (int i = 0; i < n; i++) {
        sum += a;
        temp = a;
        a = b;
        b = temp + b;
    }
    printf("Sum of Fibonacci series: %d\n", sum);
    return 0;
}
```

Finding factorial of a number

```
#include <stdio.h>
int main() {
    int n, i;
    unsigned long long factorial = 1;
    printf("Enter a positive integer: ");
    scanf("%d", &n);
    for(i = 1; i <= n; ++i) {
        factorial *= i;
    }
    printf("Factorial of %d = %llu\n", n, factorial);
    return 0;
}
```