Problem1

```c
#include <stdio.h>

int main() {
    int rows, i, j, k;
    printf("Enter the number of rows: ");
    scanf("%d", &rows);
    k = 0;
    for (i = 1; i <= rows; ++i, k = 0) {
        for (j = 1; j <= rows - i; ++j) {
            printf("  ");
        }
        while (k != 2 * i - 1) {
            printf("%d ",  k+1);
            ++k;
        }
        printf("\n");
    }
    return 0;
}
```

Problem2

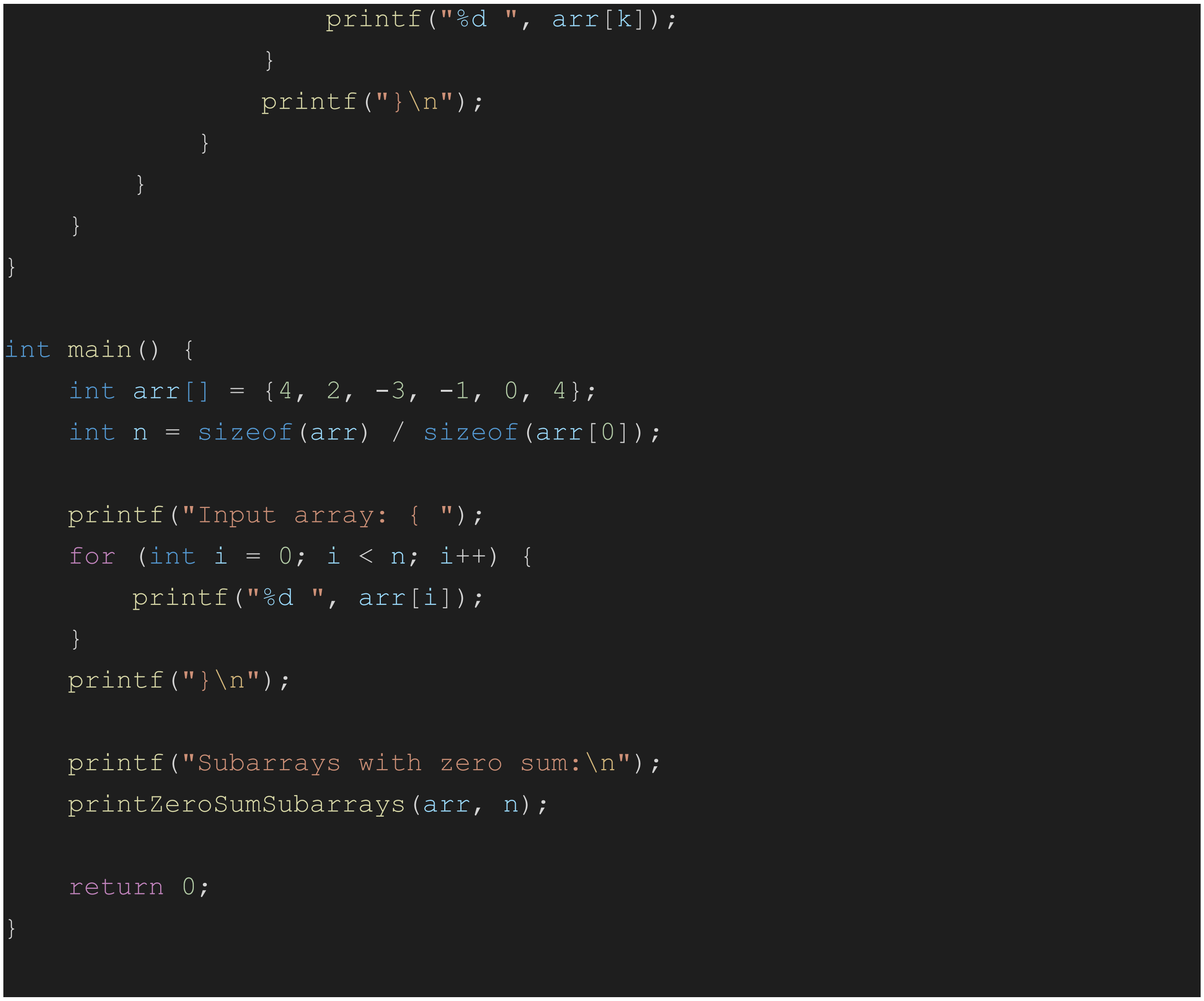
```c
#include<stdio.h>
#include<string.h>

int romanToInt(char* s) {
    int values[7] = {1, 5, 10, 50, 100, 500, 1000};
    char symbols[7] = {'I', 'V', 'X', 'L', 'C', 'D', 'M'};
    int res = 0;
    int prev_val = 0;
    int curr_val = 0;

    for (int i = strlen(s) - 1; i >= 0; i--) {
        for (int j = 0; j < 7; j++) {
            if (symbols[j] == s[i]) {
                curr_val = values[j];
                break;
            }
        }
        if (curr_val < prev_val) {
            res -= curr_val;
        } else {
            res += curr_val;
        }
        prev_val = curr_val;
    }

    return res;
}

void main(){
    char str[10];
    printf("Enter a string: ");
    fgets(str, sizeof(str), stdin);
    printf("%d",romanToInt(str));
}
```

Problem3

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

void replaceNonOverlapping(char *str, char *pattern, char replaceChar) {
    int strLen = strlen(str);
    int patLen = strlen(pattern);
    int replaceLen = 1; // length of replacement character
    int i = 0;

    while (i < strLen) {
        if (strncmp(&str[i], pattern, patLen) == 0) {
            // match found, replace non-overlapping occurrences
            for (int j = i + patLen; j < strLen; j += patLen) {
                if (strncmp(&str[j], pattern, patLen) == 0) {
                    i = j;
                    break; // overlapping occurrence found, break loop
                }
                str[j] = replaceChar;
                replaceLen++;
            }
        }
        i += patLen;
    }

    // pad string with null characters to remove trailing characters
    for (int j = strLen; j > strLen - replaceLen; j--) {
        str[j] = '\0';
    }
}

int main() {
    char str[] = "ABCABCXABC";
    char pattern[] = "ABC";
    char replaceChar = '@';

    printf("Input string: %s\n", str);
    printf("Pattern to replace: %s\n", pattern);
    printf("Character to replace with: %c\n", replaceChar);

    replaceNonOverlapping(str, pattern, replaceChar);

    printf("Output string: %s\n", str);

    return 0;
}
```

Problem4

```
#include <stdio.h>

void printZeroSumSubarrays(int arr[], int n) {
    // iterate over all subarrays
    for (int i = 0; i < n; i++) {
        int sum = 0;
        for (int j = i; j < n; j++) {
            sum += arr[j];

            // check if subarray sum is zero
            if (sum == 0) {
                // print subarray
                printf("{ ");
                for (int k = i; k <= j; k++) {
```



```c
                    printf("%d ", arr[k]);
                }
                printf("}\n");
            }
        }
    }
}

int main() {
    int arr[] = {4, 2, -3, -1, 0, 4};
    int n = sizeof(arr) / sizeof(arr[0]);

    printf("Input array: { ");
    for (int i = 0; i < n; i++) {
        printf("%d ", arr[i]);
    }
    printf("}\n");

    printf("Subarrays with zero sum:\n");
    printZeroSumSubarrays(arr, n);

    return 0;
}
```