



"Ss. Cyril and Methodius" University in Skopje

**FACULTY OF COMPUTER
SCIENCE AND ENGINEERING**

Lesson 11

Example midterm 2 problems

Structured Programming

Problem 1

Given array $A = a_i$, where $0 \leq i < n$, from integers (size and the elements) read from SI. Write a program that will form new array $B = b_i$, where $0 \leq i < \frac{n}{2}$ if n is even, or $0 \leq i < \frac{n}{2} + 1$ if n is odd, so $b_0 = \text{sum of digits of odd positions of number } a_0 = \text{sum of digits on even positions of number } a_n - 1$. Print the array b on SO. The least significant digit is on position 1. **The sum of digits on even or odd positions should be computed in a separate recursive functions (or one function).**

Problem 1

Solution

```
#include <stdio.h>

int sum_digits(int n, int odd) {
    if(odd == 1) {
        if(n < 10) return n;
        return n % 10 + sum_digits(n / 100, odd);
    } else {
        if(n < 100) return n / 10;
        return (n / 10) % 10 + sum_digits(n / 100, odd);
    }
}

int main() {
    int a[100];
    int n;
    scanf("%d", &n);
    int i;
    for(i = 0; i < n; ++i) {
        scanf("%d", &a[i]);
    }
    int b[100];
    for(i = 0; i < n / 2 + n % 2; ++i) {
        b[i] = sum_digits(a[i], 1) - sum_digits(a[n - 1 - i], 0);
    }
    for(i = 0; i < n / 2 + n % 2; ++i) {
        printf("%d\t", b[i]);
    }
    return 0;
}
```

Problem 2

Given a square matrix (the size and the elements are read from SI). Write a program that will print "YES" if the sum of elements that are at the same time over the main and under the supportive diagonal and under the main and over the supportive is larger than the sum of the elements at the same time under the main and under the supportive diagonal and over the main and over the supportive. In the opposite case print "NO".

Problem 2

Solution 1/2

```
// second problem
```

Problem 2

Solution 2/2

Problem 3

Given a text file with name "text.txt". Write a program that will print how many rows have less than average words per row. In the file the words are composed only of alphanumerical characters and the max size of a row is 80. Print the result on the SO.

Problem 3

Solution 1/2

```
#include <stdio.h>

int words(char *str) {
    int in_word = 0;
    int count = 0;
    while(*str) {
        if(isalnum(*str)) {
            in_word = 1;
        } else {
            if(in_word == 1) {
                count++;
            }
            in_word = 0;
        }
        str++;
    }
}
```

Problem 3

Solution 2/2

```
    if(in_word) count++;
    return count;
}

int words2(char *str) {
    char last = str[strlen(str) - 1];
    printf("END: %d %c", last, last);
    int count = 0;
    while(*(str + 1)) {
        if(isalnum(*str) && !isalnum(*(str + 1))) {
            count++;
        }
        ++str;
    }
    return count;
}

int main() {
    FILE* f;
    f = fopen("text.txt", "r");

    char row[200];
    int total = 0;
    int rows = 0;
    while(fgets(row, 200, f) != NULL) {
        int wc = words2(row);
        rows++;
        printf("%d : %d\n", rows, wc);
        total += wc;
    }
}
```

Materials and Questions

Lectures, exercises and announcements
`courses.finki.ukim.mk`

Source code of all examples and problems
`https://github.com/tdelev/SP/tree/master/latex/src`

Questions and discussion
`forum.finki.ukim.mk`