Tutorial 4

Problem 1:

Input 10 numbers into an array and perform the following operations:
- calculate and print the average
- print out those values above the average
- print out how many values are above the average.

e.g.: 12, 13, 45, 23, 34, 76, 56, 43, 97, 34

printout: The average is: 43.3
The numbers above the average are:
45, 76, 56, 97
There are 4 numbers above the average!

Step-by-step solution:

1. Write a function that reads from the keyboard 10 numbers
2. Write a function that receives as parameter an array, computes and returns the average. It also prints the average.
3. Write a function that receives as parameters an array and the average value and prints out:
   a. the items that have values above the average,
   b. how many items were printed
4. Write the main () function and call the implemented functions.
Solution:

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

/* function declarations */
/* read integers from keyboard */
void read_numbers(int array[], int size);

/* compute the average of the numbers from the array */
float comp_avg(int array[], int size);

/* print all numbers higher than the average */
void print_elems_above_avg (int array[], int size, float avg);

/* main function */
int main ()
{
    int numbers[10];
    float average;

    read_numbers(numbers, 10);

    average = comp_avg(numbers, 10);

    print_elems_above_avg (numbers, 10, average);

    printf("\nGoodbye! ");

    return(EXIT_SUCCESS);
}
```
/* function definitions */

/* read integers from keyboard */
void read_numbers(int array[], int size)
{
    int i;

    for (i = 0; i < size; i++)
        {
        printf("Enter the \%d-th item: ",i);
        scanf("\%d", &array[i]);
        }
}

/*compute the average of the numbers from the array */
float comp_avg(int array[], int size)
{
    int i;
    float avg;

    avg =0.0;

    for (i = 0; i < size; i++)
        {
        avg=avg+array[i];
        avg = avg/size;
        printf("The average is: %f \n", avg);
        return avg;
        }
/* print all numbers higher than the average */

void print elems above avg (int array[], int size, float avg)
{
    int noitems, i;
    noitems=0;
    printf("Numbers above the average are: ");

    for (i = 0; i < size; i++)
    {
        if(array[i]>avg)
        {
            printf("%d, ",array[i]);
            noitems++;
        }
    }
    printf("\nThere are %d items > %f", noitems, avg);
}

Problem 2:

Write a program that checks whether a word represented as a STRING is a “palindrome” or not. A “palindrome” is written in the same manner from left to right and from right to left.

   e.g. ‘radar’, “noon”, “racecar”

Note: Use the functions provided by the “string.h” library and the inverting technique
Step-by-step solution:

1. Write a function that reads a word from the keyboard
2. Write a function that inverts the word provided as parameter
3. Write a function that checks if the word and the inverted one are the same
4. Write the main () program and call the functions

Solution:

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

/* function declarations */
/* read a name as a string from the keyboard */
void read_string(char str[]);

/* check if two words are palindrome */
int is_palindrome(char str1[], char str2[]);

/* reverse the name */
void reverse_string(char str[], char rev[]);

/* main function */
int main()
{
    char word[20];
    char rev_word[20];

    read_string(word);

    reverse_string(word, rev_word);
    printf("\nThe reversed word is: %s", rev_word);
```
if (is_palindrome(word, rev_word) == 1)
    printf ("\nThe word is palindrome \n");
else
    printf ("\nThe word is NOT palindrome \n");

    return(EXIT_SUCCESS);
}

/* function definitions */

void read_string(char str[])
{
    printf("Enter your word:");
    scanf ("%s", str);
}

void reverse_string (char str[], char rev[]) 
{
    int i,j;
    j=0;

    for (i = strlen(str) -1; i>=0; i--)
    {
        rev[j] = str[i];
        j++;
    }

    rev[j] = '\0';
}

int is_palindrome(char str1[], char str2[])
{
    int i;

    if (strcmp(str1, str2) == 0)
        return 1;
    else
        return 0;
}