

RECITATION 6

Q1. Write a program that reads the temperatures of a whole week into 1 array "temperature[]" and prints the mean temperature for that week. In addition, print all days with a temperature warmer than a temperature limit. Use 3 separate functions to read the temperatures, to calculate the mean temperature and to determine which days are warmer than the user-defined temperature limit.

Q2. Write a program with following main function:

```
#include <stdio.h>
#define SIZE 12
#define COLUMNS 3

void ReadArray(int [], int);
void PrintMatrix(int [], int, int);

int main(void)
{
    int a[SIZE];
    ReadArray(a, SIZE);
    PrintMatrix(a, SIZE, COLUMNS);
    return 0;
}
```

- The function ReadArray reads 12 (SIZE) numbers and stores them in the one dimensional array a.
- The function PrintMatrix prints the 12 numbers on 4 lines of 3 (COLUMNS) numbers each.

By adapting only SIZE and COLUMNS the program must also be able to print for instance 20 numbers in a 5 x 4 matrix.

In addition, after every row, the sum of all elements in that row will be printed, and under every column, the sum of all elements in that column will be printed.

Q3. Write a program with a main function and 2 extra functions.

- In the main function an array of 100 integers is declared and the 2 extra functions are called.
- The first function reads a number of integers and stores them in the array. The user enters the integers and ends with 999 to indicate the end of the reading process. 999 can not be stored in the array!

- The second function prints the previously read integers.

The main function looks like:

```
int main(void)
{
    int row[MAX];
    int size;
    size = ReadArray(row);
    PrintArray(row, size);
    return 0;
}
```

Q4. Add an extra function to exercise Q3 that calculates the mean value of all entered integers and returns that mean value to the main function. The mean must be printed in the main function. The header of this extra function could be like:

double CalcMean(int row[], int size)

Furthermore, write a function that searches the max value of all integers entered and returns that max value to the main function, too.

Q5. Write a program with functions that merges 2 ordered rows with 5 elements (ordered from small to large) into 1 ordered row.

```
Enter ordered row: 5 9 12 54 78
Enter ordered row: 2 3 34 45 102

The merged row is: 2 3 5 9 12 34 45 54 78 102
```

Q6. Write a program with functions:

- A first function reads 20 numbers and stores them in an array.
- A second function makes sure none of the numbers in the array occurs more than once. To this end, the second, third, ... occurrence of a number is removed from the array and all other array elements are shifted to the left.

```
Enter a list of 20 numbers:
```

```
2 5 3 4 6 5 3 4 6 7 2 4 4 5 3 2 1 6 1 7
```

```
Following numbers are stored in the cleaned up array:
```

```
2 5 3 4 6 7 1
```

Q7. Write a program that generates 6 different random numbers in the interval [1, 42]. Use an array to store the numbers and to make sure all 6 numbers are different.

Q8. Write a program that simulates the rolling of 2 dice. Rolling 2 dice always results in a value between 2 and 12. How many times will every possible value occur if the dice are rolled 400 times?

- Declare an array in the main function to keep the occurrences for every possible value.
- A first function simulates 400 rolls with 2 dice and calculates the values. (Hint: `rand()%6+1` results in a random number from 1 to 6, and as such simulates the rolling of 1 die)
- A second function prints the occurrences.