

## RECITATION 4

**Q17.** Write a program with only 1 variable, that prints the numbers -3, -1, 1, 3, 5, ..., 25, comma separated to the screen. Nothing needs to be read from the keyboard.

**Q18.** Write a program that asks the user to enter an integer number and prints the multiplication table of that number. The screen dialog should look like:

```
Enter an integer number: 7
The table of multiplication of 7 is:
1 x 7 = 7
2 x 7 = 14
...
20 x 7 =140
```

**Q19.** Write a program that asks the user to enter an integer number  $n$  and prints the sum  $1+2+3+4+\dots+n$ .

**Q20.** Write a program that prints a filled square to the screen by printing 22 lines of 40 black rectangles (the ASCII code for a black rectangle is 219).

**Q21.** Write a program that first asks the user to enter an integer number. Afterwards more integers are asked until the sum of those integers equals or exceeds the first number entered. The screen dialog should look like:

```
Enter the limit: 15
Enter an integer number: 3
Enter an integer number: 4
Enter an integer number: 6
Enter an integer number: 5
The limit of 15 is reached or exceeded!
```

**Q22.** Write a program that asks the user to enter an integer number in the interval  $[-2, 4.5[$  (i.e.  $-2 \leq \text{number} < 4.5$ ) and prints it to the screen. If a wrong number is entered, a new number must be requested until a valid number is entered.

**Q23.** Write a program that asks the user to enter an integer number in the interval  $[-30, 30]$ . Make sure only valid numbers can be entered! This number is then printed as a bar graph made out of \* symbols. Negative numbers are drawn from the middle to the left, positive numbers are drawn from the middle to the right. Points are used to fill up the empty places. The entered number itself is printed in the end. Repeat the program until a 0 is entered.

```

number: 12
..... | *****
number: -4
..... **** | .....
number: 0

```

**Q24.** Write a program that sums 10 numbers entered by the user and computes the mean of those 10 numbers. Try to use only 3 variables in your program. Calculate the sum while reading the numbers.

**Q25.** Write a program that prints the mean of a number of integers. The exact number of integers is not known upfront. If the number 999 is read, the program stops reading new numbers. 999 cannot be taken into account for the calculation of the mean.

**Q26.** Write a program that reads a natural number  $n$ , calculates  $n!$  and prints the result to the screen.

- $0! = 1$
- $n! = 1 \times 2 \times 3 \times \dots \times n$  with  $n > 0$

**Q27.** Write a program that reads two numbers: a base  $b$  ( $b \in \mathbb{R}$ ) and an exponent  $n$  ( $n \in \mathbb{Z}$ ). Both positive and negative values for the exponent are possible. Afterwards, the exponentiation  $b^n$  is calculated and the result is printed to the screen. Calculate  $b^n$  with a loop. Do not use a standard function.

**Q28.** Write a program that prints the minimum and maximum value of 10 numbers entered by the user. Also, print when the minimal and maximal number were entered.

**Q29.** Write a program that calculates the greatest common divisor of 2 positive integers. Use Euclid's algorithm to determine the gcd. In this algorithm, the biggest number is replaced by the difference of both numbers. Repeat this until both numbers are equal. This number is the gcd. Print the gcd and all intermediate steps.

```
Enter 2 positive integer numbers: 114 90
```

```
114      90
```

```
24       90
```

```
24       66
```

```
24       42
```

```
24       18
```

```
6        18
```

```
6        12
```

```
6        6
```

```
The gcd of 114 and 90 equals 6.
```

**Q30.** Write a program that asks the user to enter a positive integer and prints all rows of consecutive positive integers with a sum equal to the first integer entered.

```
Enter an integer > 0: 87
```

```
87 is the sum of:
```

```
12 13 14 15 16 17
```

```
28 29 30
```

```
43 44
```

```
Enter an integer > 0: 64
```

```
64 is the sum of:
```

```
no solution
```