



Kristianstad
University
Sweden

DT555B Programmering i C

Lab 1 – Algorithm Development

Zandra Hedlund

230926

Table of Contents

1. Introduction	1
Task 1 - Computer Assisted Instruction (CAI).....	1
Task 2 Multiplication Table	1
2. Design	2
Task 1 - Computer Assisted Instruction (CAI).....	2
Pseudocode	2
Flowchart	3
Test cases.....	3
Task 2 Multiplication Table	4
Pseudocode	4
Flowchart	5
Test cases.....	6
4. Results and discussion	6
5. References.....	7

1. Introduction

Algorithm development is the process of designing and creating a set of step-by-step instructions or procedures to solve a specific problem or perform a particular task in programming. It involves defining the problem, understanding the requirements, and come up with a sequence of actions that can be followed to achieve the desired outcome. Algorithm development is a fundamental aspect of software engineering and data analysis, and it plays a crucial role in solving complex problems. Effective algorithm development requires a combination of creativity, problem-solving skills, and an understanding of data structures.

Task 1 - Computer Assisted Instruction (CAI)

This task was to write an algorithm for a program that will help primary school students practice addition. The program will generate an addition question, with two random numbers in the range 0-100 and the student will provide the answer. The question will be repeated until the student gives the right answer.

Task 2 Multiplication Table

This task was to write an algorithm for printing a $n \times n$ multiplication table of size n . The user will enter a number between 1-11 to generate the table. The table should be formatted with the numbers in straight right aligned columns and a border, as shown below:

```
Enter the size of the multiplication table: 6
```

x	1	2	3	4	5	6
1	1	2	3	4	5	6
2	2	4	6	8	10	12
3	3	6	9	12	15	18
4	4	8	12	16	20	24
5	5	10	15	20	25	30
6	6	12	18	24	30	36

2. Design

I started by breaking down the task into smaller tasks, to figure out which steps were needed to get the desired result. I then used pen and paper to make a step-by-step instruction, which I then translated into pseudocode and a flowchart. I used a top-down design, since I think that makes it easy to follow the logic. I tried to install AlgoBuild, but it didn't work on my Macbook, so I used an online tool, lucid.app to make the flowcharts.

Task 1 - Computer Assisted Instruction (CAI)

My algorithm starts with assigning two integer variables, a and b, which will be assigned two randomised numbers, using a randomising function, like rand(). The program will then enter a while loop, that will run as long as the students answer is not the same as the sum of the two numbers. The student will be asked to add the two numbers together and input an answer. The program will compare the students answer to the sum of the two generated numbers. If the students answer and the sum of the two numbers are identical, the program prints "Very good!" and the exit. If the answer is wrong, the loop starts over and the student gets the same question again, until the answer is correct and the while condition evaluates to false.

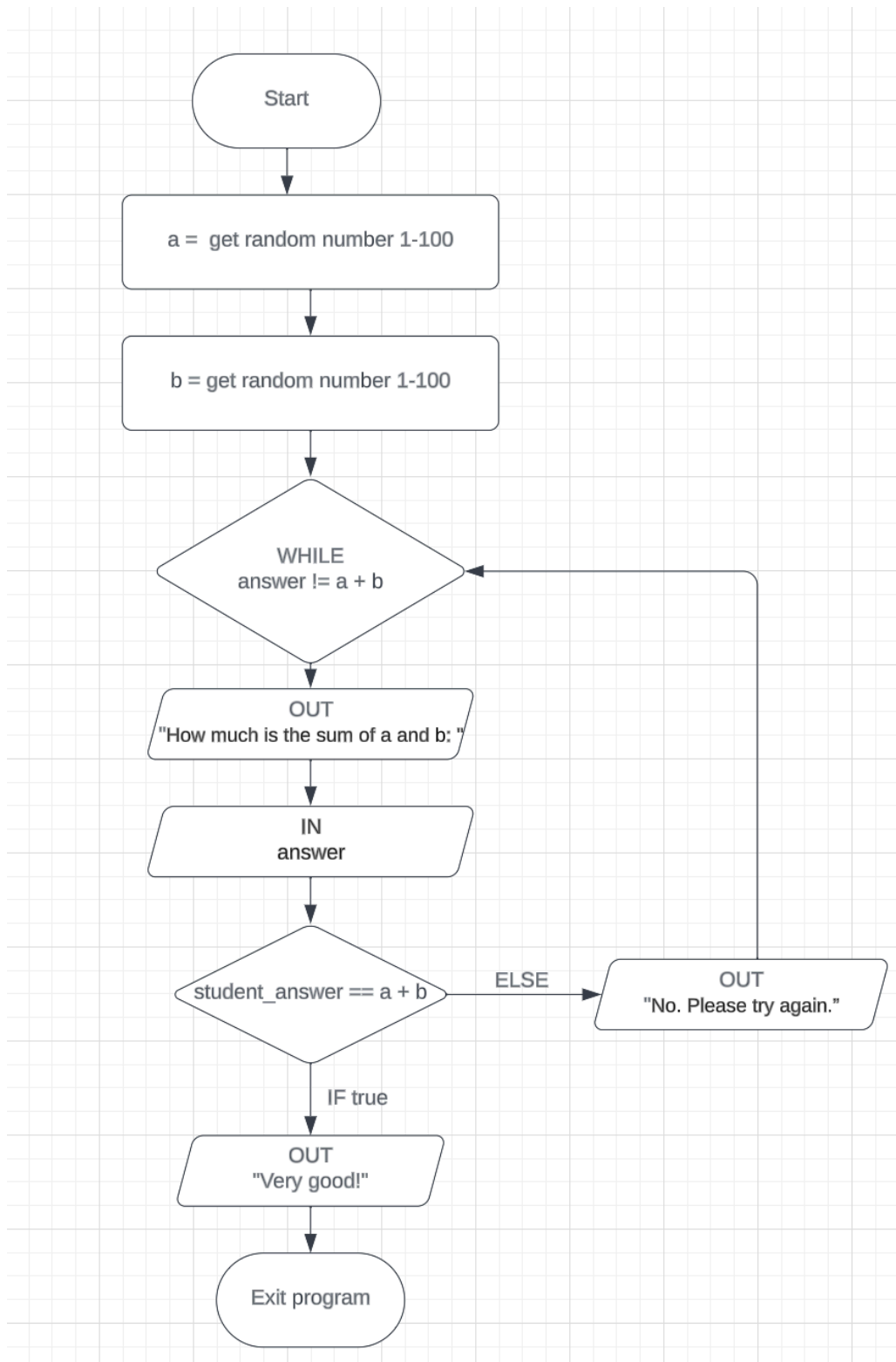
Pseudocode

a = get a random number in the range 0—100

b = get a random number in the range 0—100

```
WHILE answer != a+b:
    OUT "How much is the sum of a and b: "
    IN answer
    IF answer == a+b
        OUT "Very good!"
        Exit program
    ELSE
        OUT "No. Please try again"
END WHILE
END PROGRAM
```

Flowchart



Test cases

Input both right and wrong answers and check that the program makes the correct calculations and behave as expected.

Task 2 Multiplication Table

My algorithm starts by printing a statement asking the user to input a number. If the number is less than one or larger than 11, the program will print an error message and exit. If the number is in the correct range, it will start with printing the header row. This is done using a for loop, that will run n times to print the numbers in the header. To get the formatting correct, it should start with printing " x |" before the loop, but since the subject said that we didn't have to consider how to produce the output format, I left that out of the flowchart. I also chose not to include how to print the columns with the right spacing and alignment. After the header is printed, it will print a newline and another for loop is used to print the separator line. To print the table row by row, I'm using a nested for loop. The outer loop prints the row label on the left side of the table and the inner loop calculates and prints each number. After each iteration of the inner loop it should also print a newline to continue on the next line, but I accidentally left that out of the flow chart. When all numbers are printed, the program exit.

Pseudocode

OUT "Enter the size of the multiplication table: "

IN number

IF number <1 or >=11

 OUT "Error message"

 Exit program

ELSE

 FOR i = 1; i <= n; i ++;

 OUT print header row

 END FOR

 OUT print newline

 FOR i = 1; i <= n; i ++;

 OUT print separator line

 END FOR

 OUT print newline

 FOR i = 1; i <= n; i ++;

 OUT print row label

 FOR j = 1; j <= n; j ++;

 calculate number: i * j

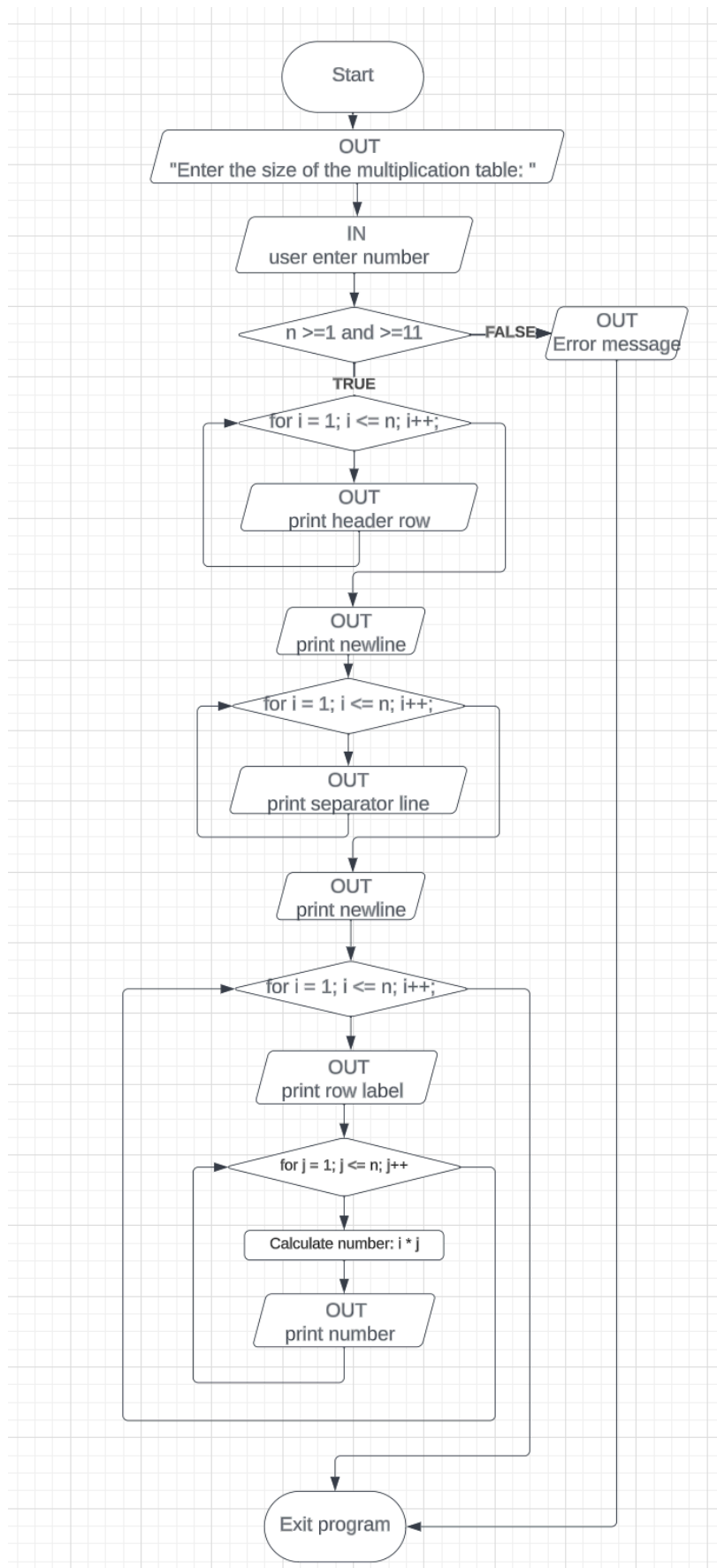
 OUT print number

 END FOR

 OUT print newline

END PROGRAM

Flowchart



Test cases

Test with different number inputs and check that the calculations is correct. Make sure the formatting stays the same, even when it prints 3 digit numbers, as will be the case with multiplication table 10 and 11.

4. Results and discussion

The hardest part was making the flowcharts in digital format, every program I tried was either not compatible with my computer or was extremely buggy. It took a very long time making them, which was very annoying since that wasn't really the main task. Besides that, I think it was a good exercise. Algorithms are of course essential in programming and learning to plan ahead before diving into the actual coding, really makes things easier later on.

5. References

I didn't use any external references, besides the lectures provided on Canvas.