**What is Algorithm?**

* Algorithm is a *range of logical steps* to solve a problem and *to reach a specific goal*.
* All steps for solving a specific problem.
* The most *important step of programming* is making an *algorithm* / developing.
* Coding an algorithm in a programming language is an easier part.

There are 5 base criteria of Algorithm:

1. **Efficiency**

Computers can’t think. That’s why every step of algorithm must be understandable, simple and accurate. It must not include comments and any other needless elements. Must not include meaningless repetitions and can be used in other algorithms.

1. **Finite**

Every algorithm should consider starting point, specific steps and ending point. It must not loop infinitely.

1. **Accuracy**

Result must be accurate, and must produce same results for same data.

1. **Input / Output**

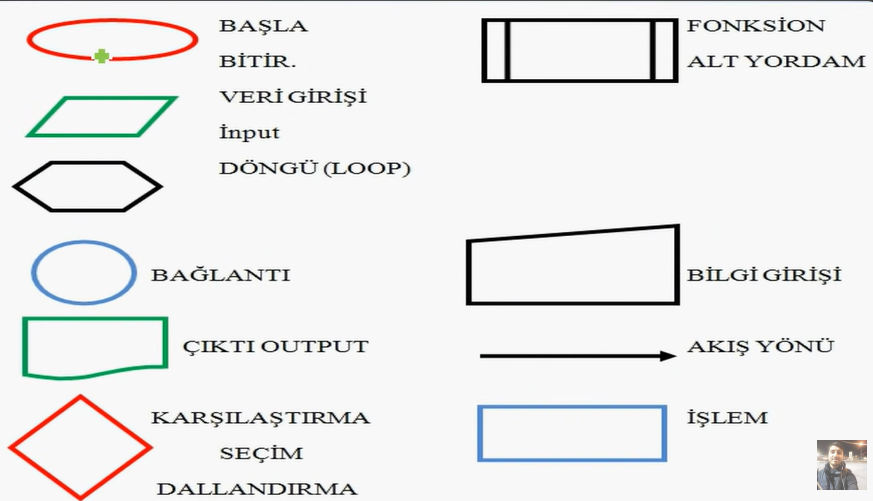
Algorithm has input values for working on it and output values for our production.

1. **Success / Performance**

Goal is to write programs that can deal with memory, efficient with duration and other performance criteria.

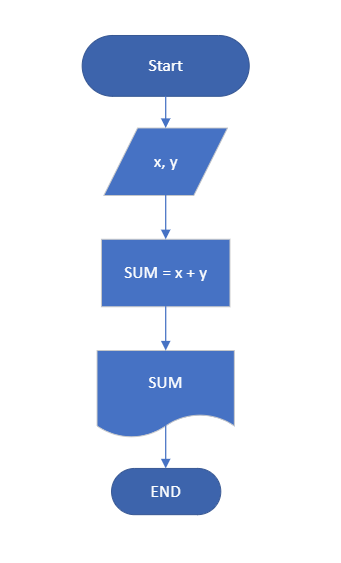
**Flowcharts**

Expressing algorithm as visualization and symbols is *flowcharts.* Difference between flowcharts and algorithm is steps in here are written in symbols and flows are shown with arrows.



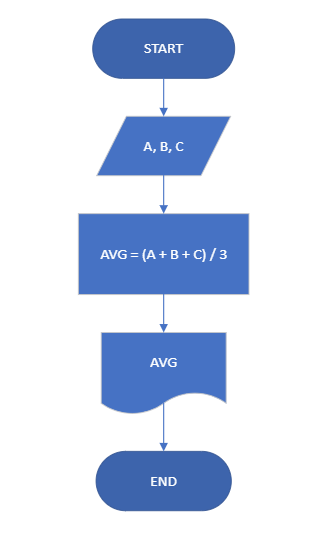
Sum of 2 numbers that is inputted from keyboard as a text algorithm and flowchart:

1. Start
2. Input 2 numbers (x, y)
3. Sum of x and y (sum = x+y)
4. Print “sum”
5. Stop



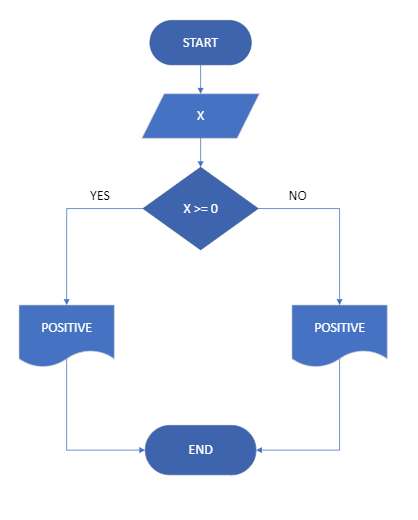
Algorithm for calculating average of a student from inputting 3 exam scores.

1. Start
2. Input exam scores (A, B, C)
3. AVG of exam scores (AVG = (A+B+C)/3)
4. Print AVG
5. Stop



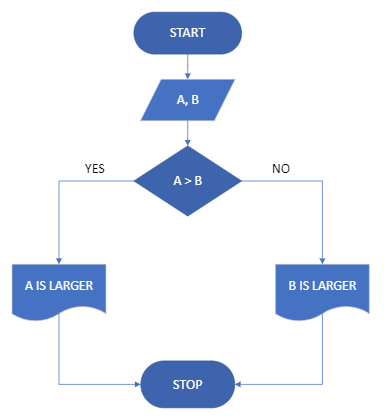
Algorithm and flowchart to check if the keyboard input is negative or positive?

1. Start
2. Input a number from keyboard (X)
3. X>=0, situation = ‘positive’
4. Else, situation = ‘negative’
5. Print
6. Stop

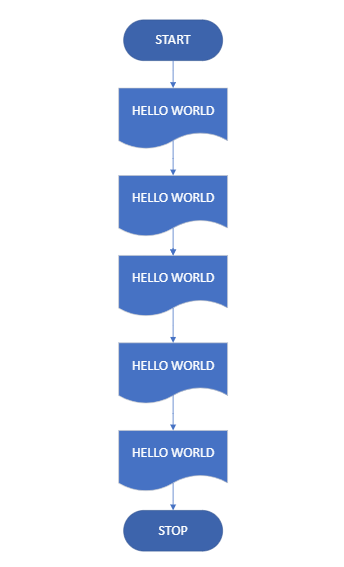


Print larger number between 2 inputted numbers – their algorithm and flowchart?

1. Start
2. Input 2 numbers (A, B)
3. If A>B, large = A
4. Else, large = B
5. Print large
6. Stop

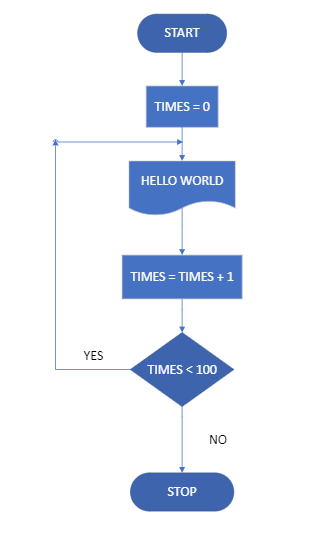


Flowchart of a message that says “Hello world” 5 times in monitor?



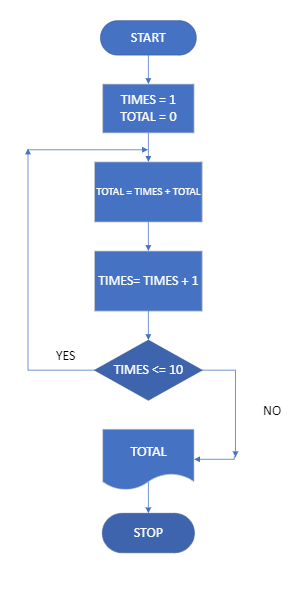
Flowchart of a message that says “Hello world” **100** times in monitor??

1. Start
2. Times = 0
3. Print “Hello world”
4. Times = times +1
5. Times <100, go to step 3
6. Else, go to step 7
7. Stop

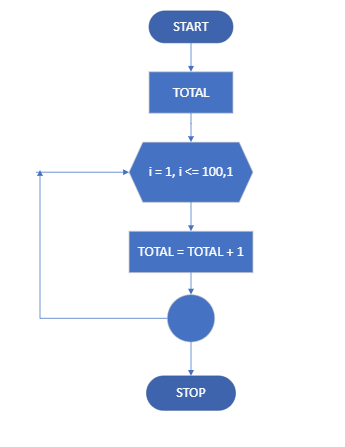


Flowchart of program that calculates the sum of numbers between 1 and 10 (including 1 and 10).

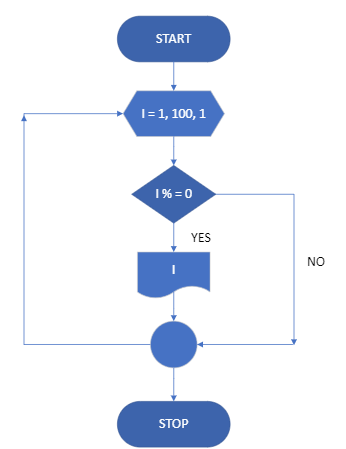
|  |  |
| --- | --- |
| TIMES | TOTAL |
| 1 | 1 |
| 2 | 3 |
| 3 | 6 |
| 4 | 10 |
| 5 | 15 |
| 6 | 21 |
| 7 | 28 |
| 8 | 36 |
| 9 | 45 |
| 10 | 55 |



Flowchart for sum of numbers from 1 to 100?

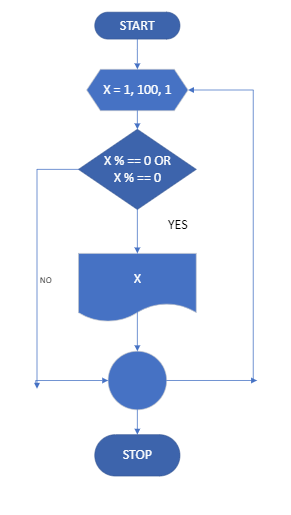


Flowchart of program printing numbers that can be divided to 8 between 1 and 100.

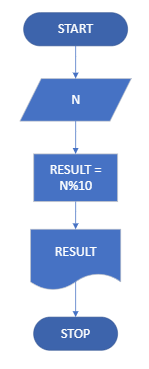


PRACTICAL EXERCISES

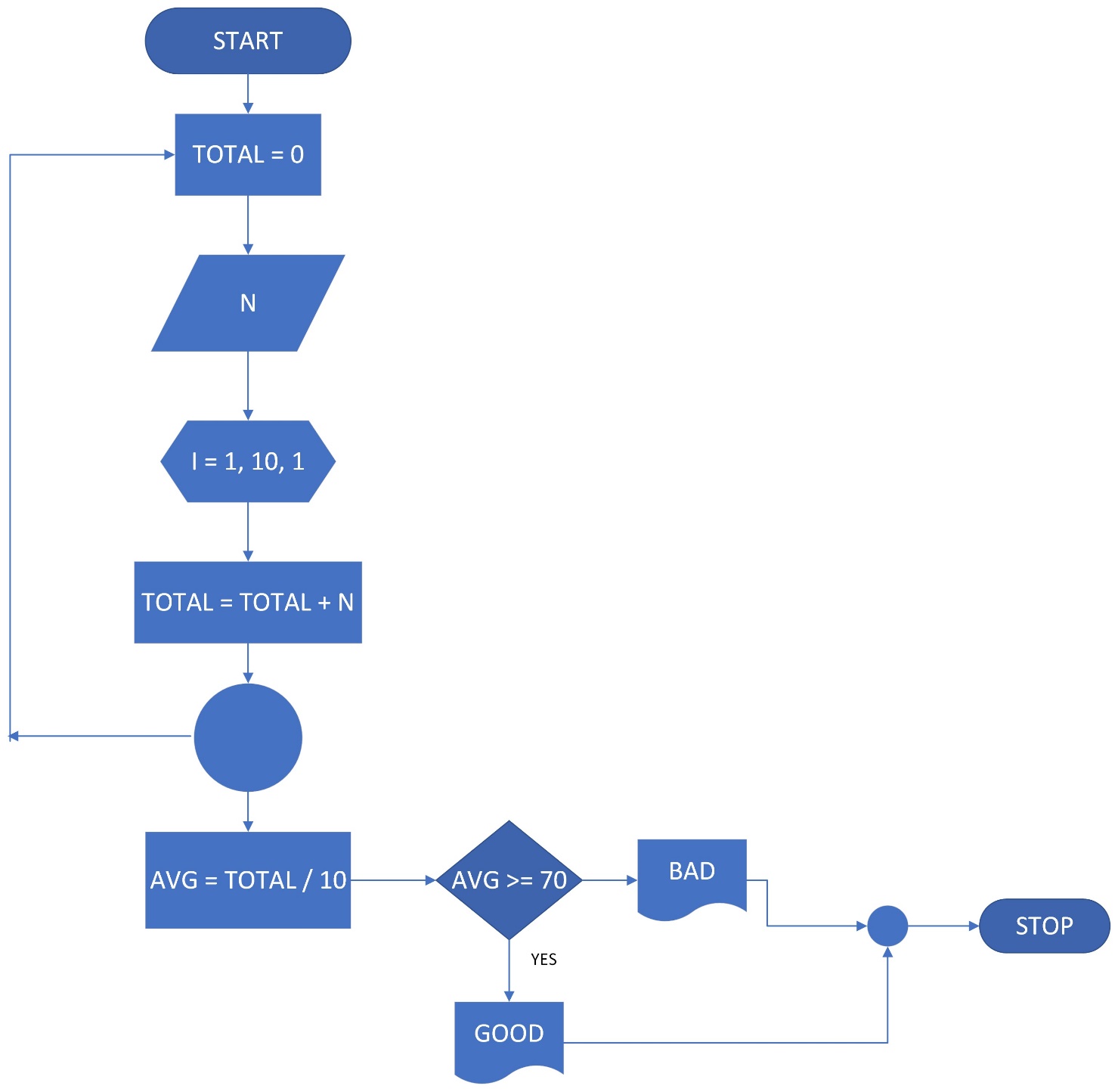
1. Flowchart of numbers that can be divided to 3 **OR** 5, from 1 to 100?



1. Flowchart of program that shows each number in a inputted number?



1. Flowchart for average of scores of students from exam, where it is good if average higher than 70, if it is lower than 70 is bad.
2. Start
3. Total = 0
4. Exam score as input
5. If number of students inputted is 10, go to step 6
6. Else go step 3
7. Total = total + exam score (S)
8. Average = total/10
9. If average >= 70, good
10. Else average < 70, bad
11. Print Average
12. Stop



**FIBONACCI SERIES FLOWCHART AND ALGORITHM (first 10 element)**

1; 1; 2; 3; 5; 8; 13; 21; 34; 55;

1. Start
2. A = 1; B = 1
3. Print A, B
4. I = 1, 8, 1
5. C = A + B
6. Print C
7. A = B
8. B = C
9. Stop

A B C

--- --- ---

1 1 2

1 2 3

2 3 5

3 5 8

