1. Start with stating/understanding the problem that you want an algorithm to solve.
2. What ingredients or tools are available for the algorithm to call on to solve the problem?

Variables, assignment operations, \( I = I + 1 \), etc.

The problem today:

A. Output: histogram, table, rows which is a

B. Median: that number in \( 50, \ldots, 100 \) to row:

so half of the N grades are \( \leq m \).

The hand, eye, electric pencil algorithm demonstrated!

For each grade \( g \):

raise an exception and throw it away if it is not one of the integers 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10

otherwise,

locate the bin labelled with grade \( g \),

write a new dot in \( B[g] \),

then done with the inputs, figure out the number of students with each grade by counting the dots in each bin.
Arrays are an important tool for use in algorithms to be implemented in the most common computer programming systems.

To use an array in Java, TWO things must be coded:

1. **Declare a variable that will eventually but not yet refer to the array.**
   ```java
   int gradeArray[];
   int histogram[];
   ```
   *Type of each element*  
   *Name of the variable that will refer to the array*  
   *Empty square brackets indicate that the variable will be a variable whose values the type int.*

2. **Within executable code, code**
   ```java
   gradeArray = new int[nStudents];
   histogram = new int[11];
   ```
   *operations like these to CONSTRUCT the array and assign the reference to the array as the value of the variable declared in step 1.*

   ```java
   histogram = new int[11];
   ```
   *means construct something*
   *the name of the variable whose value refers to the array after the assignment.*