







Computing > AP®/College  
Computer Science  
Principles > Algorithms >  
Building algorithms  
**Building algorithms**

-  The building blocks of algorithms
-  Practice: The building blocks of algorithms
-  **Expressing an algorithm**
-  Practice: Expressing an algorithm

Next lesson  
Evaluating algorithms

Computing • AP®/College Computer Science Principles • Algorithms • Building algorithms

# Expressing an algorithm

AP.CSP: AAP-2 (EU), AAP-2.A (LO), AAP-2.A.2 (EK), AAP-2.A.3 (EK)

Created by Pamela Fox.

 Google Classroom    Facebook    Twitter    Email

We can express an algorithm many ways, including natural language, flow charts, pseudocode, and of course, actual programming languages.

Natural language is a popular choice, since it comes so naturally to us and can convey the steps of an algorithm to a wide audience. When we're developing algorithms, we are often working both with people who know programming and those who don't—but they all know natural language.

However, natural language has its drawbacks. It has a tendency to be ambiguous and too vaguely defined, since it has no imposed structure. That makes it difficult for others to follow the algorithm and feel confident in its correctness. Flow charts and pseudocode are more structured formats that

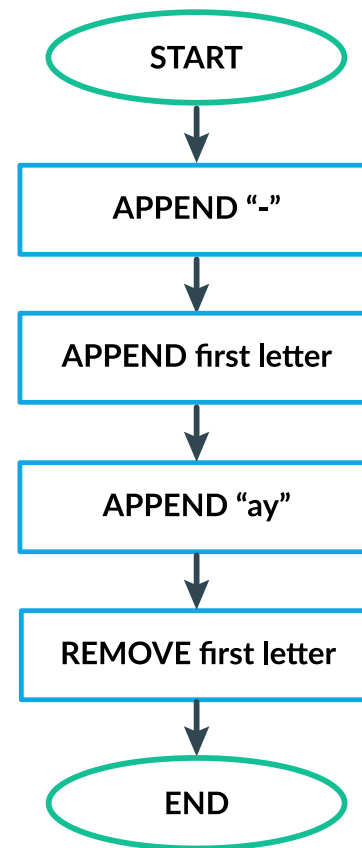
can more precisely express an algorithm, and are popular with computer scientists and programmers.

Let's try out the more structured formats by expressing the Pig Latin algorithm from the previous article.

## Flow charts

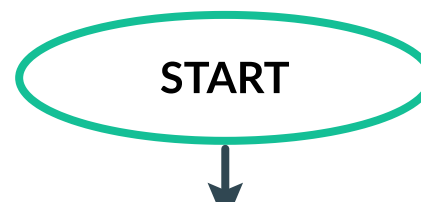
A more formal way to express an algorithm is with a **flow chart**, a diagram with boxes connected by arrows.

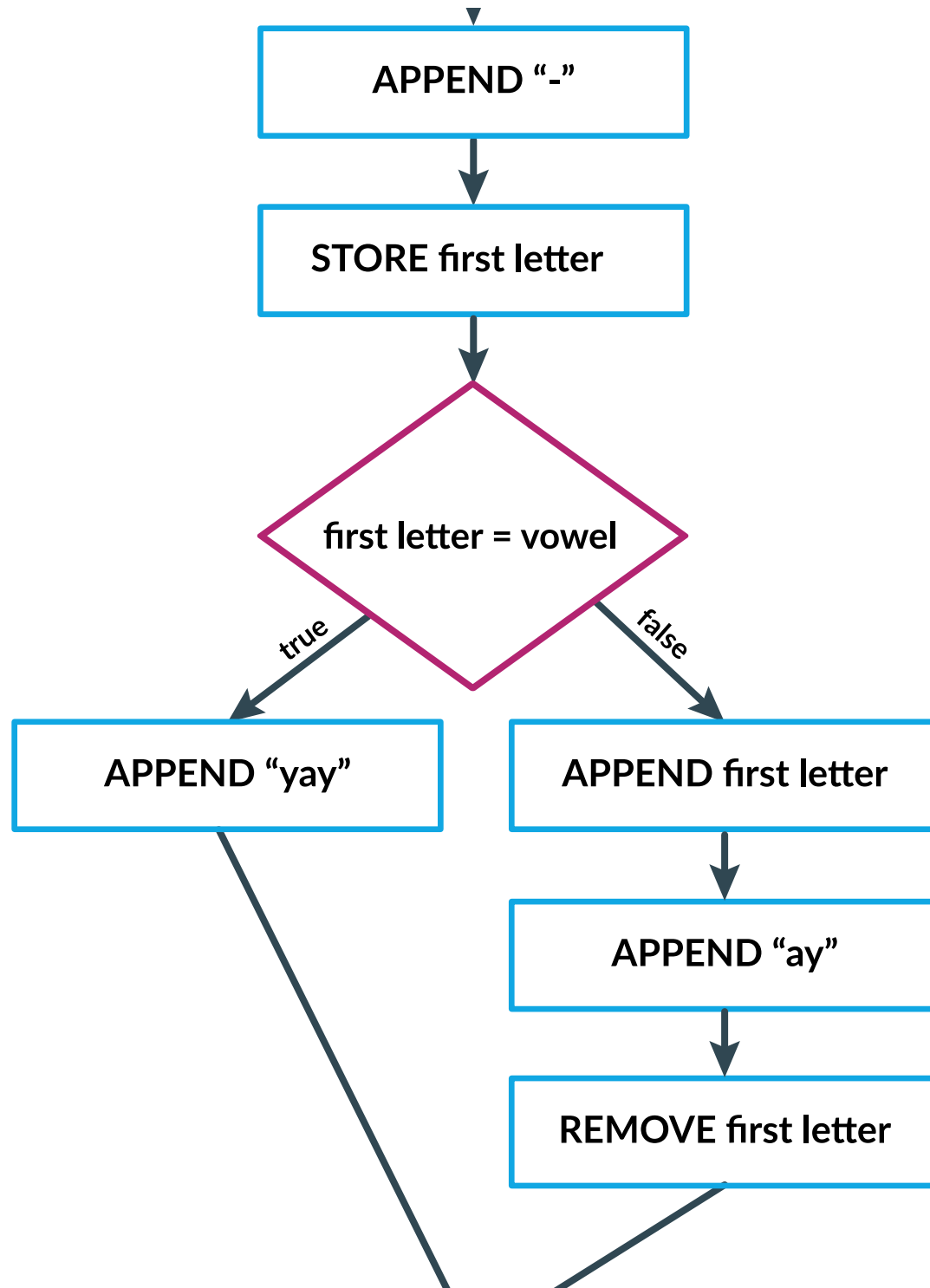
To start simple, here's a flow chart for the basic version of the Pig Latin algorithm:

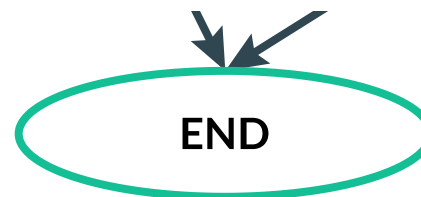


Each rectangle represents a step in the sequence, and the arrows flow from one step to the next.

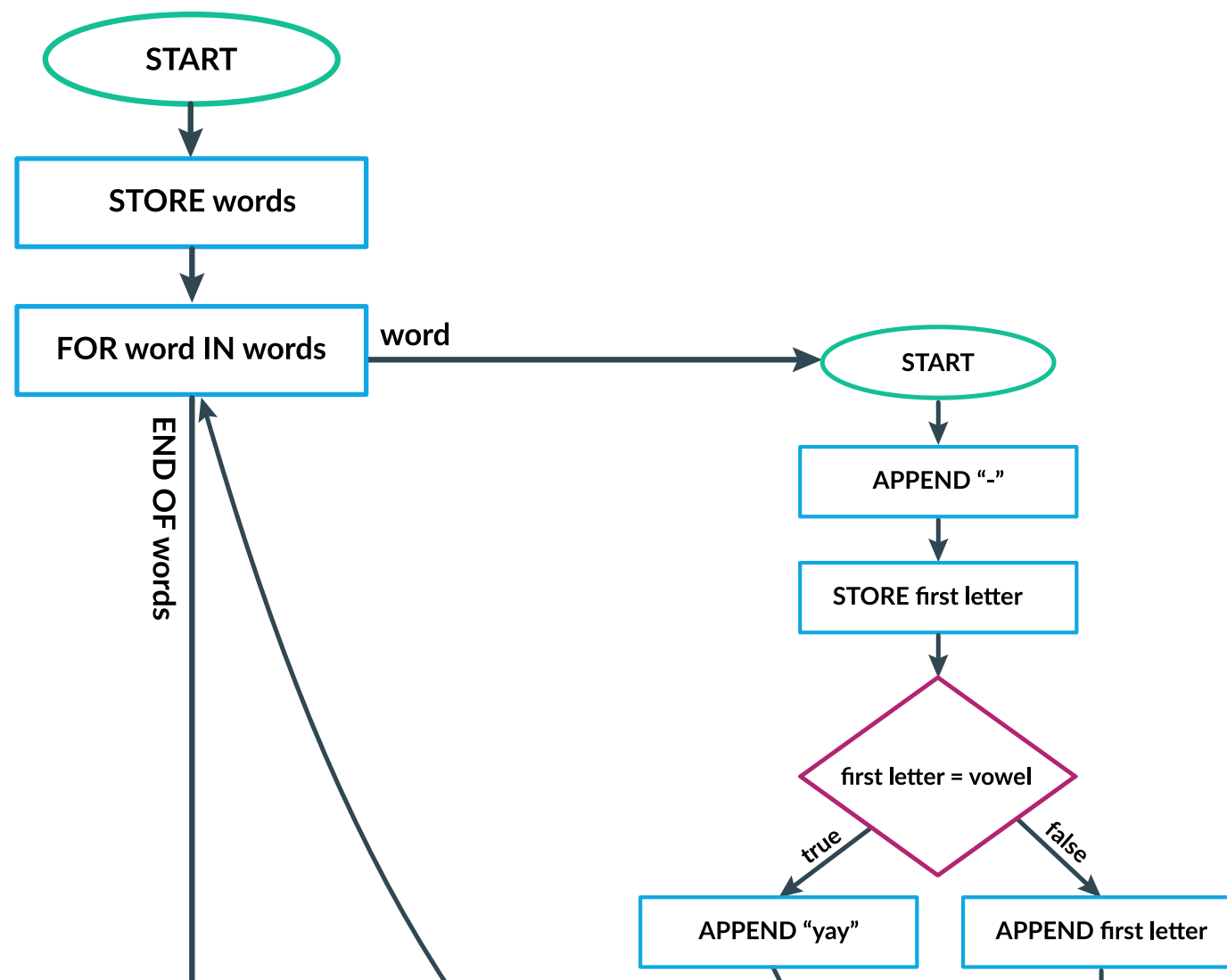
This next flow chart is for the improved algorithm and uses a diamond to represent the selection phase:

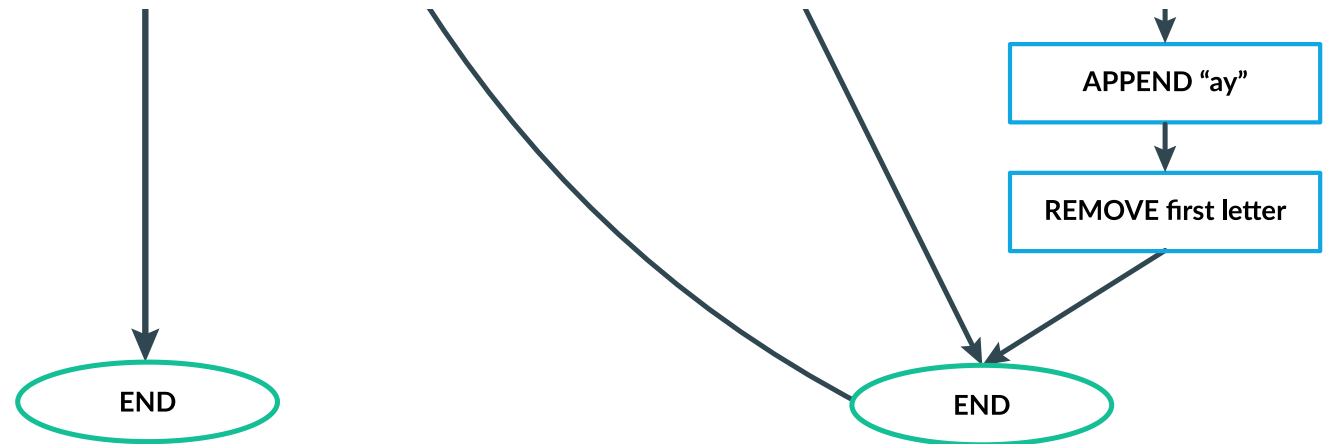






Finally, this flow chart visualizes the complete algorithm with iteration:





Expressing an algorithm in a flow chart allows us to visualize the algorithm at a high level, plus it forces us to think very carefully about sequencing and selection. Which arrow goes to what node? Are there missing arrows? Those are the kinds of valuable questions that can come up while translating an algorithm into a flow chart.

## Pseudocode

Ultimately, most algorithms become code that actually runs on a computer. Before that happens, programmers often like to express an algorithm in **pseudocode**: code that uses all the constructs of a programming language, but doesn't actually run anywhere.

Here's the Pig Latin algorithm written in the style of AP CSP pseudocode:

```
FOR EACH word IN words
{
  APPEND(word, "-")
  letter ← FIRST_LETTER(word)
  IF (IS_VOWEL(letter)) {
    APPEND(word, "yay")
  } ELSE {
    APPEND(word, letter)
    APPEND(word, "ay")
    REMOVE_FIRST(word)
  }
}
```

Every programmer writes pseudocode differently, since there is no official standard, so you may run into pseudo-code that looks very different.

Expressing an algorithm in pseudocode helps a programmer think in familiar terms without worrying about syntax and specifics. It also gives computer scientists a language-independent way to express an algorithm, so that programmers from any language can come along, read the pseudo-code, and translate it into their language of choice.

## Programming languages

Once we've planned our algorithm, whether in natural language, flow charts, pseudocode, or a combination, it's time to turn it into running code.

We'll translate the Pig Latin algorithm to JavaScript, since that's our language of choice on Khan Academy. We can use a `for` loop for the repetition, an `if/else` for the selection, and then a mix of string and array operations for the steps.



```
1 // Returns true if the given character is
  an English vowel
2 function isVowel(char) {
3     var vowels = ['a', 'e', 'i', 'o', 'u'];
4     return vowels.indexOf(char.toLowerCase
5     ()) !== -1;
6 }
7 // Pig Latin algorithm
8 var words = ["peanut", "butter", "and",
9 "jelly"];
10 for (var i = 0; i < words.length; i++) {
11     var word = words[i];
12     word += "-";
13     var firstLetter = word.charAt(0);
14     if (isVowel(firstLetter)) {
15         word += "yay";
16     } else {
17         word += firstLetter;
18         word += "ay";
19         word = word.slice(1);
20     }
21     words[i] = word;
```

eanut-pay, utter-bay, and-yay, elly-jay

[View in new tab](#)See similar code in: [App Lab](#) | [Snap](#) | [Python](#)

An algorithm can be translated into any general purpose programming language. For proof, just check out [RosettaCode.org](#), a wiki that lists hundreds of algorithms translated to over 700 languages.

Software is written in a variety of programming languages, depending on what it does and who's making it. Khan Academy currently uses four different languages for different parts of the codebases. Fortunately, we can use the same algorithms everywhere!



Do you have any questions about this topic? We'd love to answer—just ask in the questions area below! [\[Where is it?\]](#)

---

Sort by: Top Voted ▼

Questions

Tips & Thanks

---

Want to join the conversation?

Log in



Devan Eilbert a year ago

more ▼

Why does Khan Academy use 4 different programming languages? Why not consolidate to one language?

Reply • Comment

(4 votes) ▲ Upvote ▼ Downvote 🚩 Flag more ▼



pamela ♥ a year ago



more ▼

There are different reasons for each language. We use JavaScript for the frontend, since JavaScript is the language understood by web browsers. On the backend, we currently use a mix of Python and Kotlin. However, we are moving the backend over to Go, for performance reasons. So we will have a mix of those three languages for a while, since there is a **lot** of code to upgrade!

(Also, our data analytics team uses a mix of Python and R, two languages that have a lot of available libraries for data processing)

Comment

(10 votes) ▲ Upvote ▼ Downvote 🚩 Flag more ▼



Fire 2 years ago

more ▼

Algorithms are what are used to create path-finding, correct?

Reply • Comment

(4 votes) ▲ Upvote ▼ Downvote 🚩 Flag more ▼



pamela ♥ 2 years ago

more ▼

Yes, that's correct, there are various algorithms that can be used for path-finding.

Here's a fun visualization of different algorithms:

<https://qiao.github.io/PathFinding.js/visual/>

<https://www.redblobgames.com/pathfinding/a-star/introduction.html>

2 comments

(7 votes)  Upvote  Downvote  Flag **more** 



more ▾

Reply • Comment

(5 votes)  Upvote  Downvote  Flag **more** 



more ▾

Q) Create a flowchart for:

Reply • Comment

(2 votes)  Upvote  Downvote  Flag **more** 



more ▾

This is what I would do if I understand what you're asking correctly.  
(where input is the function's input) (periods were used just for spacing)

```
.....\<<<<<<<<<<<<<<
.....\>...../\ no
Start -> N = input -> F = 0 -> F = F + 1/N -> N = N - 2 -> if N ≤ 0
.....\> yes
```

```
.....V
.....return F
```

Try to go through the flowchart to understand how it works. Hope this makes sense! (:

**Comment**

(2 votes)  Upvote  Downvote  Flag **more** 



**NAVEED RIAZ** 6 months ago

**more** 

So whatever v were doing previously, making small executable programmes using variables, strings, procedures, v were actually making algorithms, right? The steps to go about achieving a result.

**Reply • Comment**

(1 vote)  Upvote  Downvote  Flag **more** 



**Martin** 6 months ago

**more** 

Strictly speaking, yes. Those programs give the computer a set of instructions to follow to achieve a particular result.

**Comment**

(2 votes)  Upvote  Downvote  Flag **more** 



**786737566** 6 months ago

**more** 

Is 4 the limited number of languages?

**Reply • Comment**

(1 vote)  Upvote  Downvote  Flag **more** 



**Martin** 5 months ago

**more** 

No according to the article, Khan Academy uses four different languages

for different parts of the site. That's specific to this site, there are hundreds of programming languages. But you can take algorithm and implement it in hundreds of different programming languages, in that way algorithms are universal. Although some languages are better at some algorithms than others.

Comment

(1 vote)  Upvote  Downvote  Flag [more](#) 



**layaz7717** 2 months ago

[more](#) 

What does `!= -1` mean (it's in the code example)?

Reply • Comment

(1 vote)  Upvote  Downvote  Flag [more](#) 



**KLaudano** 2 months ago

[more](#) 

"`vowels.indexOf(char.toLowerCase()) != -1`" compares whether "`vowels.indexOf(char.toLowerCase())`" does **not** equal -1. This would be the opposite of "`vowels.indexOf(char.toLowerCase()) === -1`" which would determine if the expression does equal -1.

3 comments

(1 vote)  Upvote  Downvote  Flag [more](#) 



**Ayah Bakri** 9 months ago

[more](#) 

i cant understand creating or using programming languages and applying them to create an algorithm! can someone please help? thank you

Reply • Comment

(1 vote)  Upvote  Downvote  Flag [more](#) 

[< The building blocks of algorithms](#)[Expressing an algorithm >](#)

AP® is a registered trademark of the College Board, which has not reviewed this resource.