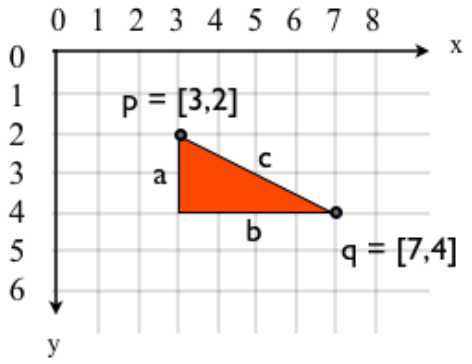
Intro to programming II

Week 6

The distance between two points

This is about the most useful formula in video games so let's see it in detail. We want to find the distance **c** between two points **p** and **q**. We can split the vector between them into a vertical component with a magnitude of **a** and a horizontal component with a magnitude of **b**.

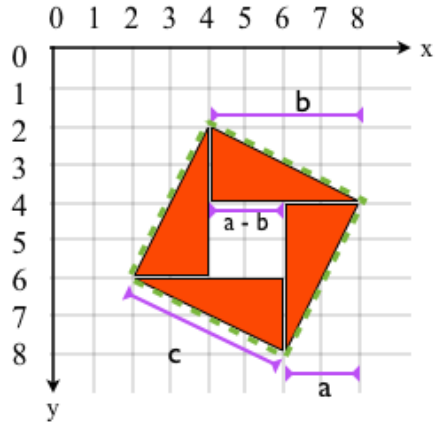


We know that **a** and **b** are simply the absolute value of the difference of the rows and columns of p and q. In our example:

$a = |p_y - q_y| = |2 - 4| = |-2| = 2$
$b = |p_x - q_x| = |3 - 7| = |-4| = 4$

where the absolute value is the function that makes its argument positive., i.e., |-4| = 4.

Now we need to express **c** in terms of **a** and **b**. This relationship is given by the Pythagorean theorem, which follows from the following image:



We know that the area of a rectangle is the product of the lengths of its sides:

area(□) = side1 x side2

Likewise, the area of a triangle is half its base by its height:

area(△) = (base x height) / 2

We can write the area **A** shown by the green dotted line in two ways. We can either, multiply its sides:

$A = c^2$

or we can add the area of the white square inside the red triangles to the area of the 4 triangles:

$A = (a - b)^2 + 4\ (a\ b / 2)$
$= a^2 - 2ab + b^2 + 2ab$
$= a^2 + b^2$

Since these are 2 ways of expressing the same area A we have:

$c^2 = a^2 + b^2$

i.e., the square of the hypotenuse is equal to the sum of the squares of the sides.

From here we find that the value of **c** is

$c = \text{sqrt}(a^2 + b^2)$

In our example, this means that the distance between **p** and **q** is

$c = \text{sqrt}((p_x - q_x)^2 + (p_y - q_y)^2)$

or

$c = \text{sqrt}(2^2 + 4^2)$
$= \text{sqrt}(20)$
$= 4.47$

Intro to programming II

http://www.codeskulptor.org/#examples-memory_template.py

Week 6

List methods

Methods is the name of functions that are part of an object. Some useful list methods:

```
my_list = [3, 6, 2, -5]
```

- does the element exist in the list?

```
>>> 2 in my_list
true
```

- add an element to the end of the list

```
>>> my_list.append[7]
>>> print(my_list)
[3, 6, 2, -5, 7]
```

– find the index of an element

```
>>> my_list.index(-5)
3
>>> my_list.index(10)
  File "<stdin>", line 1
    my_list.index(10))
                      ^
SyntaxError: invalid syntax
```

- remove the first element

```
>>> my_val = my_list.pop()
>>> print(my_list)
[6, 2, -5, 7]
>>> print(my_val)
3
```

– remove the element of index i

```
>>> my_list = [3, 6, 2, -5]
>>> my_val = my_list.pop(1)
>>> print(my_list)
[3, 2, -5]
>>> print(my_val)
6
```

- remove the element of value x

```
>>> my_list = [3, 6, 2, -5]
>>>my_list.remove(-5)
>>>print(my_list)
[3, 6, 2]
```

Iteration

There are two iteration processes that are used all the time.

mapping: it turns all the elements of an array into something else.

```
def square(a):
    result = []
    for idx in range(len(a)):
         result.append(a[idx] * a[idx])
    return result
```

or, accessing the elements directly:

```
def square(a):
    result = []
    for elem in a:
        result.append(elem * elem)
    return result
```

filtering: we keep only certain elements of the array.

```
def positives(a):
    result = []
    for elem in a:
        if elem > 0:
            result.append(elem)
    return result
```

Dictionaries

A dicitonary associates a single name, called the 'key', to many values:

```
>>> my_dictionary = {"peter": 10, 2015: "year"}
>>> print( my_dictionary["peter"]
10
>>> print(my_dictionary[2015])
year
```

To iterate over a dictionary:

```
for key, value in my_dictionary.items():
    print(key, value)
```