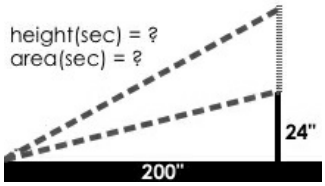
Top Down / Bottom Up

A retractable flag pole starts out 24 inches tall, and grows taller at a rate of 0.6in/sec. An elastic is anchored 200 inches from the base and attached to the top of the pole, forming a right triangle. Using a top-down or bottom-up strategy, define functions that compute the *height* of the pole and the *area* of the triangle after a given number of seconds.



Directions: Define your first function (`height` or `area`) here.

Contract and Purpose Statement

Every contract has three parts…

`#` `area` `::` `Number` `->` `Number`

function name | *domain* | *range*

`#` Consumes seconds & produces the area of the triangle with a base of 200 and changing height

what does the function do?

Examples

Write some examples, then circle and label what changes…

`examples:`

`area ( 5 )` **`is`** `1/2 * (200 * height(5))`

function name | *input(s)* | *what the function produces*

`area ( 6 )` **`is`** `1/2 * (200 * height(6))`

function name | *input(s)* | *what the function produces*

`end`

Definition

Write the definition, giving variable names to all your input values…

`fun` `area( sec ):`

function name | *variable(s)*

```
1/2 * (200 * height(sec))
```

what the function does with those variable(s)

`end`

Directions: Define your second function (`height` or `area`) here.

Contract and Purpose Statement

Every contract has three parts…

`#` `height` `::` `Number` `->` `Number`

function name | *domain* | *range*

`#` Consumes the # of seconds and produces the height, according to h=0.6s + 24

what does the function do?

Examples

Write some examples, then circle and label what changes…

`examples:`

`height ( 1 )` **`is`** `(0.6 * 1) + 24`

function name | *input(s)* | *what the function produces*

`height ( 2 )` **`is`** `(0.6 * 2) + 24`

function name | *input(s)* | *what the function produces*

`end`

Definition

Write the definition, giving variable names to all your input values…

`fun` `height( sec ):`

function name | *variable(s)*

```
(0.6 * sec) + 10
```

what the function does with those variable(s)

`end`