Bisection Method

Write the **R** function `bisection.r` that calculates the root of the user-defined function f on a closed interval $[A, B]$ using the Bisection Method.

Inputs:

1. A - the left endpoint of the interval.
2. B - the right endpoint of the interval.
3. t - a user-defined tolerance

Output: x - a close approximation of a root of the function f on the closed interval between a and b.

Basically, the algorithm calculates the midpoint M between A and B. If $f(M)$ has the same sign as $f(A)$, then the root must be between M and B. Therefore, set A equal to M. On the other hand, if $f(M)$ does not have the same sign as $f(A)$, then the root must be between A and M. In this case, set B equal to M. Repeat.

How long do you repeat the algorithm?

You continue while $|B - A| > t$ and $\frac{2\times|B-A|}{|A|+|B|} > t$.

You must check to see whether $f(A)$ and $f(B)$ have opposite signs. If they have the same sign (so there is no guarantee that there is a root between A and B), then return `NA` and print an error message.

Good luck. Here are several examples. For the first two examples, $f(x) = \cos(x) - 0.80 + 0.10x^2$. For the third example, $f(x) = -\sin(x) + x/50$. And, for the last two examples, $f(x) = (x-3)^5$.

```
> bisection(0,pi,0.001)
[1] 0.7267234
> bisection(0,pi,0.000001)
[1] 0.7270473
> bisection(pi/2,pi,0.0001)
[1] 3.079994
> bisection(0,5,0.0001)
[1] 2.999992
> bisection(0,2,0.0001)
Error, f(A) and f(B) must have different signs.
[1] NA
```