# Solution to Problem 1

**Graded by:** Group 1 (Ya Chen, Saku Kukkonen, Prabir Yadav)
**Adopted from a solution by:** Saku Kukkonen, Prabir Yadav

## Problem Statement

You are standing on a road in Death Valley, California. The temperature is 119F (48.3C), and you have no water. You do know that water is located somewhere along the road, but you are disoriented from the heat, and do not know in which direction. You have no vehicle – you have to walk to find the water. Assuming that the water is located at distance x from you, give an algorithm for finding the water where you are sure to walk a total distance of no more than cx, for some constant c. In writing up your solution, be sure to state the value of c associated with your solution and remember: you do not know the value of x.

## Algorithm

```
seek_water()
begin
   d:=0;
   found:=false;
   for i:=1 to (not found) do
        begin
           olddistance:=d;
           walk( p(i), d, found );// d and found are output parameters
           i:=i+1;
        end
   // so found it
   abs_x := d - olddistance;
   return d/abs_x;
end
```

```
walk(p, d, found)
begin
   // walk does like this
   if 0 <= x <= p ∨ p <= x <= 0
      then
            increment d by x and
            set found to true;
        else
            increment d by 2p and
            set found to false;
      end
```

```
// Function p(i) returns the direction and distance to travel from the
// starting point for step i. Assuming the road stretches in the East-West
// direction. The East is the +ve direction and West is the -ve direction


p(i)
begin
  return( (-2)**(i - 1) );
end
```

## Analysis and Solution

Let us assume that $x$ is reached at the $n+1^{th}$ iteration. In the $n^{th}$ step distance from the origin $= 2^n$ In the $n+1^{th}$ step distance from the origin $= 2^{n+1}$

So, $x$ lies between $2^n$ and $2^{n+1}$, i.e.,

$$2^n \leq |x| \leq 2^{n+1} \tag{1}$$

Now, the worst case will be when the walker finds water at a distance $|x|$ in the negative direction, i.e., the direction opposite to the direction in which he started his search.

So, in the worst case the total distance traveled,

$$\begin{aligned} D &= 2\sum_{i=0}^{n} 2^i + 2\left(2^{n+1}\right) + |x| \\ D &= 2\sum_{i=0}^{n+1} 2^i + |x| \end{aligned}$$

Now,

$$\sum_{i=0}^{n} 2^i = \frac{2^{n+1}-1}{2-1} < 2^{n+1}$$

Therefore,

$$\begin{aligned} D &< 2 \times 2^{n+1} + 2 \times 2^{n+1} + |x| \\ D &< 4 \times 2^{n+1} + |x| \\ D &< 8 \times 2^n + |x| \end{aligned}$$

Now, we know from ineqaulity 1 that $2^n \leq |x|$.

So,

$$\begin{aligned} D &< 8|x| + |x| \\ D &< 9|x| \end{aligned}$$

And therefore,

$$C = 9$$

Thus, there exists a constant $C$ such that the total distance travelled is smaller than $C|x|$.

## GradingPolicy

Thumb rule: An error or shortage takes one point.

| Points | |
|---|---|
| 0–5 | Algorithm |
| 0–5 | Analysis and Solution |