How to THINK like a Programmer

Problem Solving for the Bewildered

paul vickers

chapter 3

description languages & representations

www.cengage.co.uk/vickers
©2008 PAUL VICKERS

Purpose

- This chapter is about some of the notations and languages you can use for representing problems and finishes by introducing a semi-formal language (known as pseudo-code) that you will use for writing down your problem solutions

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

Natural language

- The language of human communication
  - English, Spanish, Chinese, Inuit, ...
- Much human communication is non verbal
  - Intonation of voice, physical gestures, timing of speech
  - The words themselves do not carry the whole meaning
- For computers to 'understand' us the words we use **must** carry the whole meaning
- Precision in expression helps to avoid making errors of the third kind: solving the wrong problem but doing it well (e.g. making excellent coffee when tea was requested)
- Words are useful for thinking about problems, but they are not the only tool

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

Alternative descriptions

- We can use diagrams, pictures, and visual thinking as alternative thinking and descriptive tools.

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

ACTIVITY

I have a canvas bag in which there are five red jelly beans and a single blue jelly bean. If I put my hand into the bag, without looking in, what is the likelihood that I will pull out a red bean? And what is the likelihood that I will get the blue one? How do you visualize this problem?

©2008, PAUL VICKERS

Jelly beans

- p(red) = 5/6, p(blue) = 1/6
- Did you use a mental list to do this?
- A mental list is ok for simple problems like this but...

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

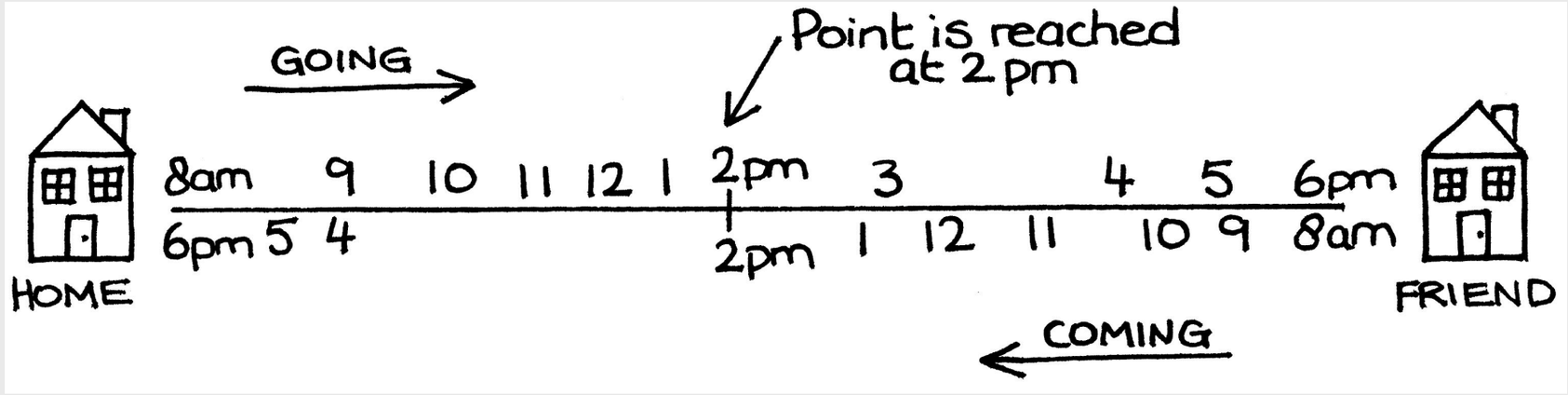
ACTIVITY

One morning at eight o'clock you set off in your car to visit a friend who lives some distance away. You encounter the odd traffic delay and your driving speed varies as the speed limits on the different stretches of road change. It is a long drive so you stop twice to refresh yourself and to eat and drink something. You arrive at 6.00 p.m. and decide to stay overnight. The next morning you set off again at 8.00 a.m. You take the same route back, but encounter heavy traffic in different places. Because one traffic jam was so large you only stop once for refreshments. Still, you arrive back home at six o'clock. **Question**: is there a point on the road that you pass at exactly the same time of day on the outward and return journeys?

©2008, PAUL VICKERS

Seeing the solution

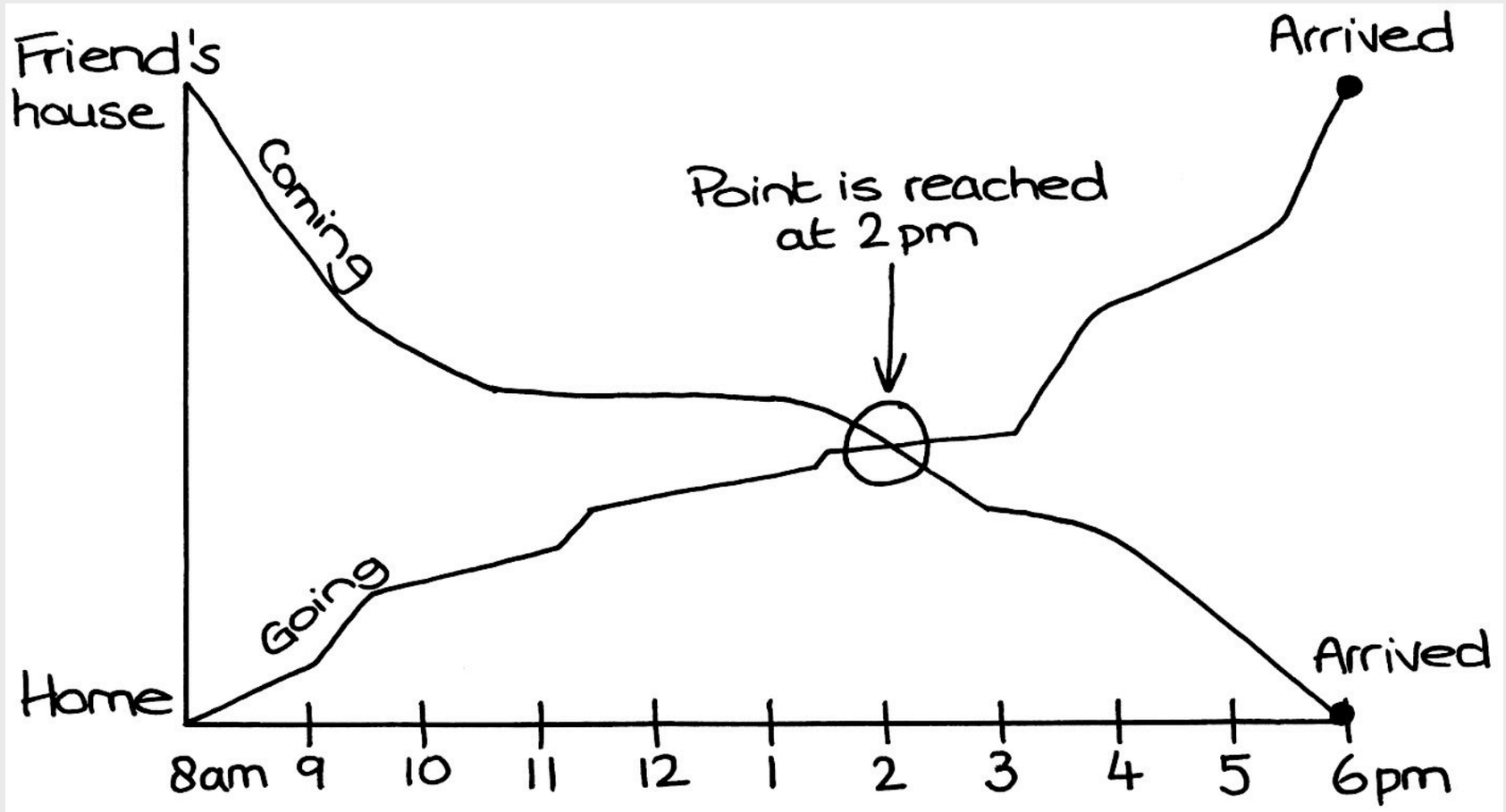
- Natural language does not lend itself to solving the problem
- A diagram helps though



©2008, PAUL VICKERS
www.cengage.co.uk/vickers

Another diagram

- Here's another way of visualizing the problem:



©2008, PAUL VICKERS
www.cengage.co.uk/vickers

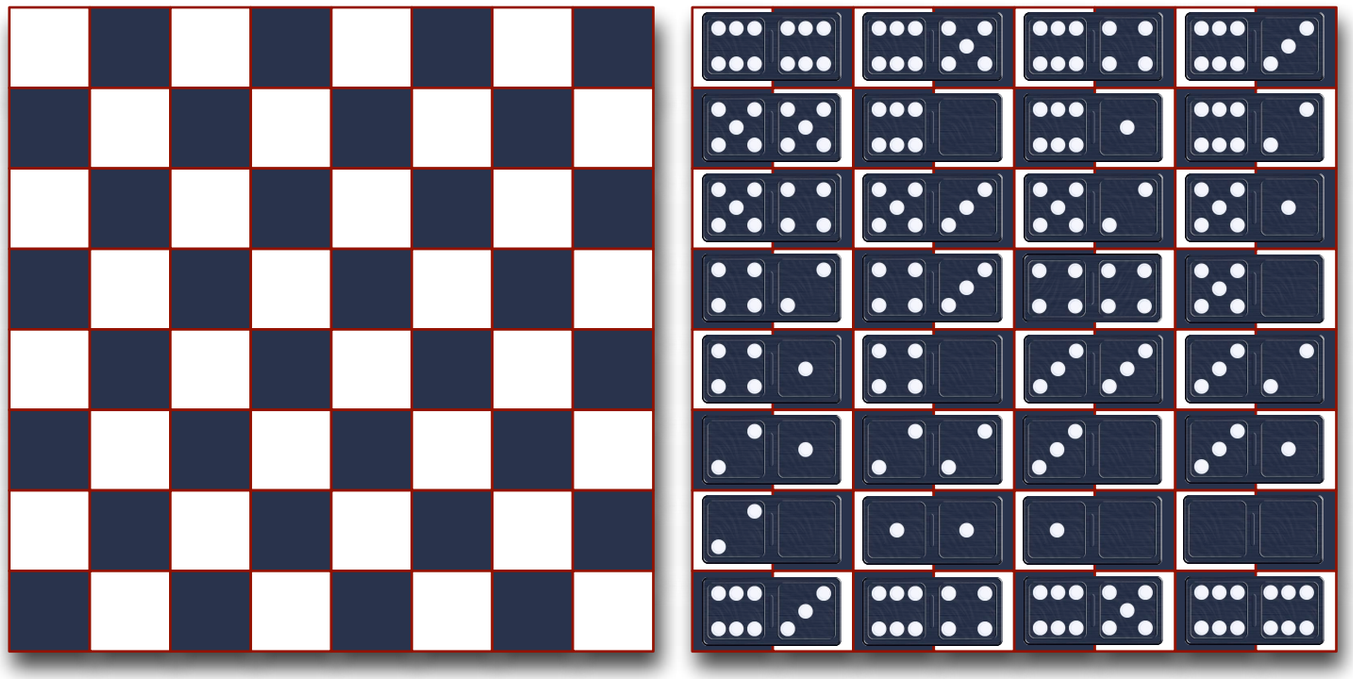
Restate the problem

- Alternatively, restating the problem can help:
- Instead of thinking of two journeys on two different days and looking for a point, just imagine you and your friend each setting out from your homes at eight in the morning. Will you meet along the road? Of course you will, so the problem is solved and there is no need for a diagram or graphs and rulers.
- By simplifying the problem (above) it can be easier to solve
- Try the following problem:

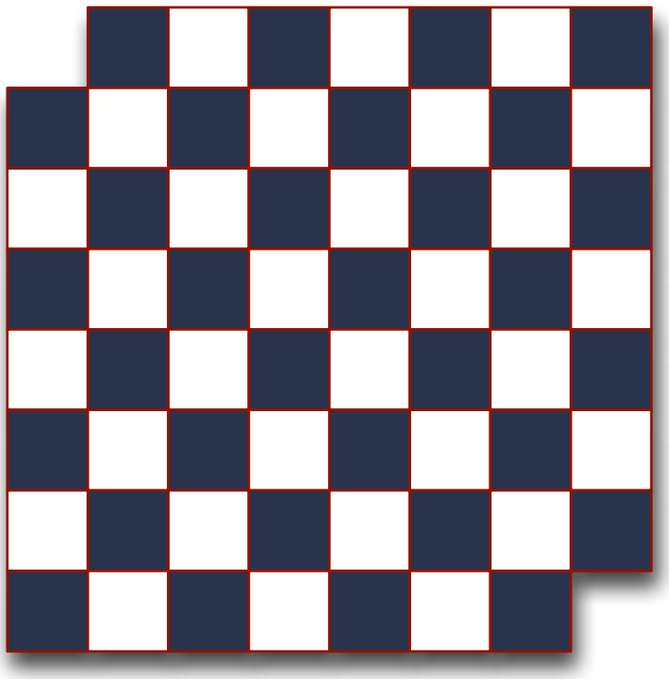
©2008, PAUL VICKERS
www.cengage.co.uk/vickers

ACTIVITY

Take a chessboard, and thirty-two dominoes. Each domino fits over two (non-diagonally) adjacent squares on the chessboard. It is easy to cover the chessboard with the dominoes (Fig. a). Now cut two opposite corners from the chessboard, leaving sixty-two squares (Fig. b) and then remove one domino. Can you still cover the chessboard?



(a)



(b)

©2008, PAUL VICKERS

Chess & dominoes

- You cannot cover the modified chess board
- A mental image here may have led you to think you could cover it
- If you only thought about the number of squares and the number of dominoes you may have got the wrong answer
- Need to consider all the details:
  - Each domino covers a white & a black square.
  - The squares removed were both white, therefore, the board can no longer be covered

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

Restate the problem

- Imagine we have thirty-two men and thirty-two women who want to get married. Instead of dominoes we now have thirty-two marriage certificates which can only be used to marry one man to one woman. All the marriage certificates can be used because we have thirty-two eligible couples. Now, remove two men (white squares). Can we use up all the marriage certificates? No, because we have thirty men and thirty-two women.
- Make sure you use all the information given

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

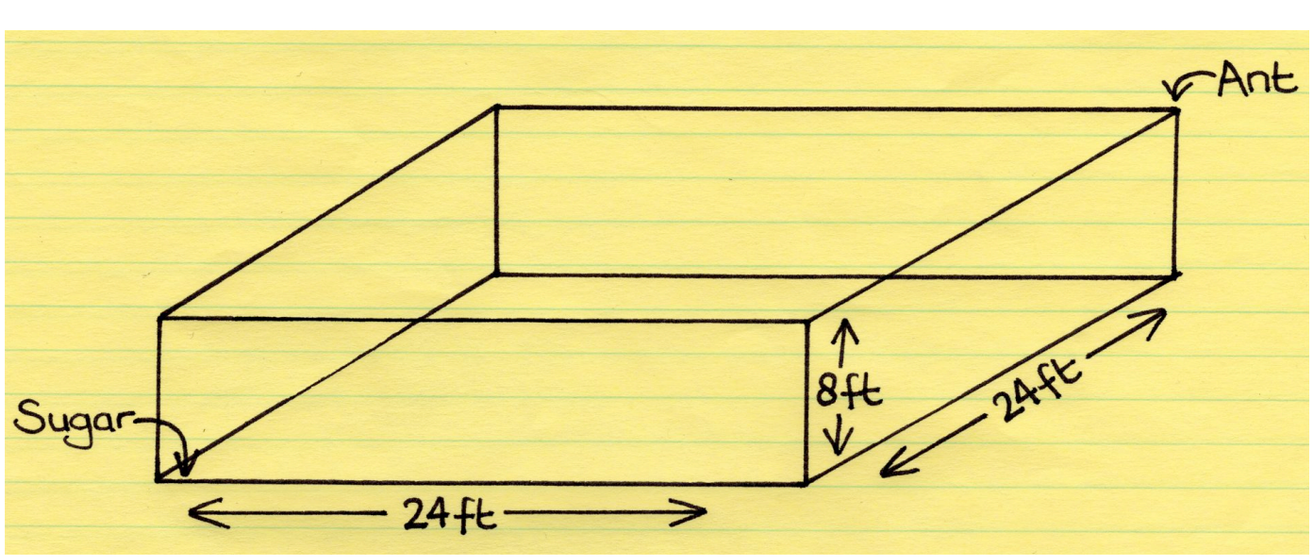
Diagrammatic folly

- Sometimes, even diagrams can lead astray
- Consider the next problem

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

ACTIVITY

There is a large square room whose walls are twenty-four feet long. The ceiling is eight feet high. On the floor in a corner is a bowl of sugar. In the opposite corner by the ceiling is an ant. What is the shortest path the ant can take to get to the sugar?

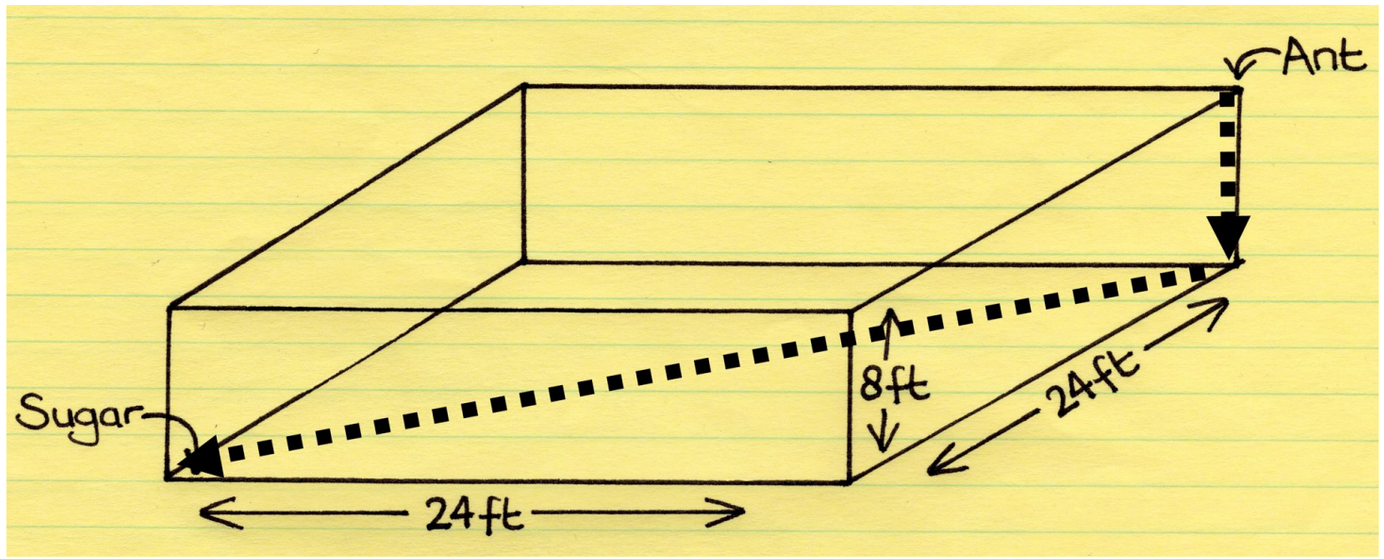


Think about it carefully and then trace the shortest walking route from the ant to the sugar

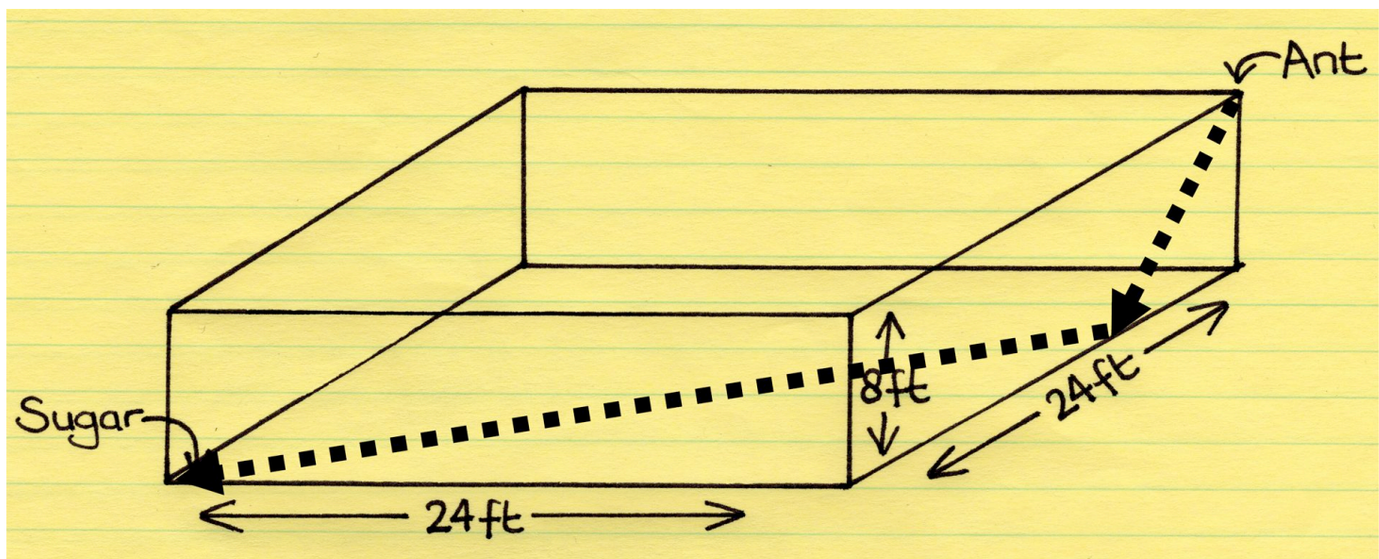
©2008, PAUL VICKERS

Solutions?

Which of the candidate solutions below is correct? A or B?



Solution A

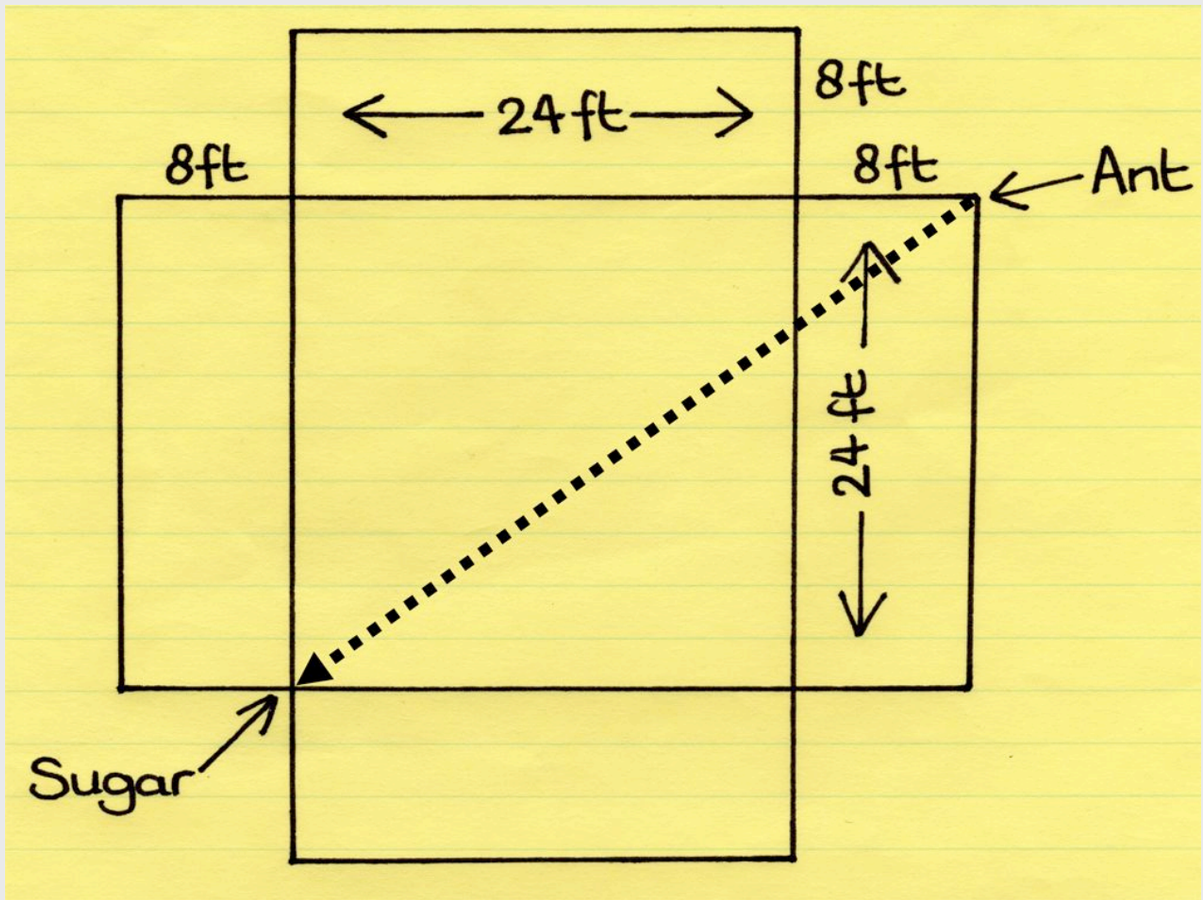


Solution B

©2008, PAUL VICKERS

Ant sugar solution

- Answer is Solution B
- We can verify this by drawing a 2D plan view rather than the 3D models above

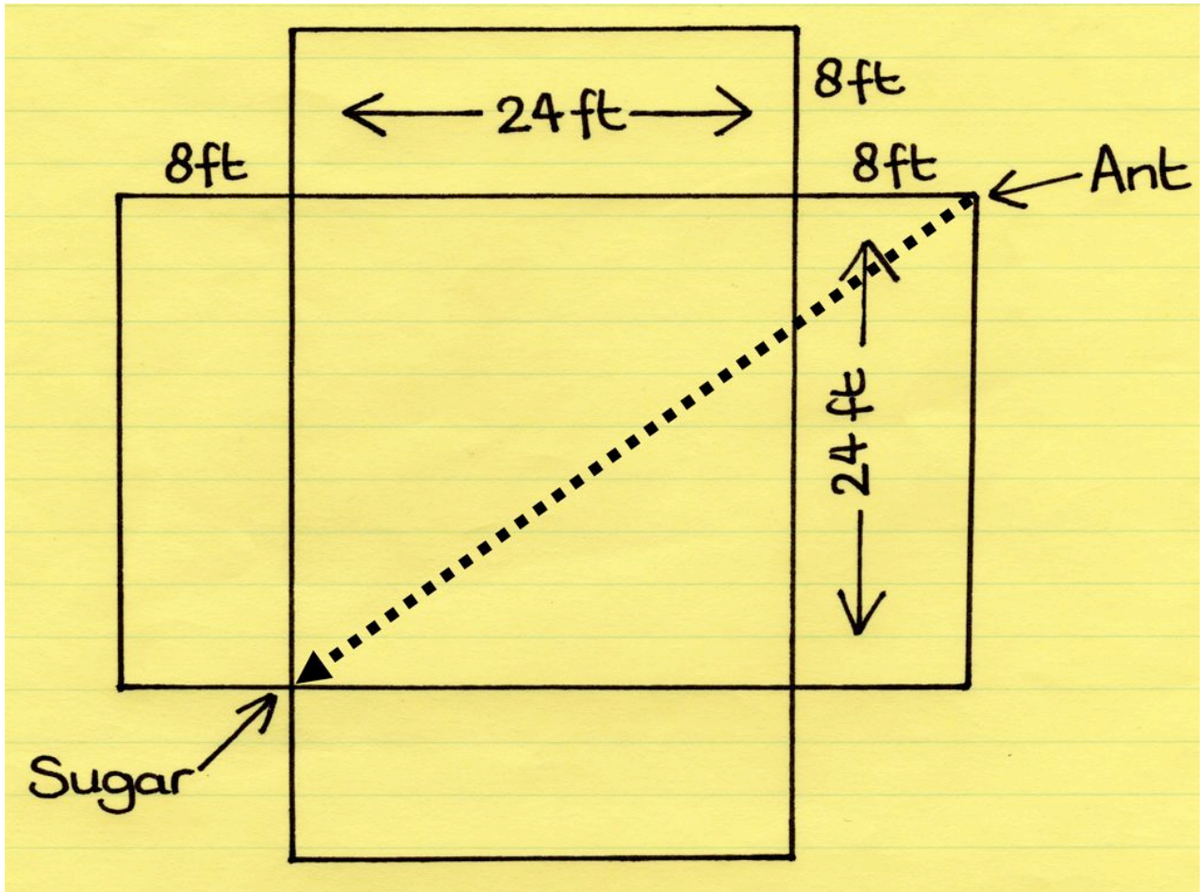


$$\text{path}^2 = (24 + 8ft)^2 + 24ft^2$$
$$\therefore path^2 = 32^2 + 24^2$$
$$\therefore path^2 = (4 \times 8)^2 + (3 \times 8)^2$$
$$\therefore path^2 = (5 \times 8)^2 \text{ :Pythagoras 3-4-5 triangle}$$
$$\therefore path = 40ft$$

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

ACTIVITY

If you still cannot see how this solution is correct, then print it out, cut out the diagram along the outside edge and fold up the four walls. You will see that you have now reproduced Solution B from slide 16



©2008, PAUL VICKERS

How many problems?

- The ant-and-sugar problem was composed of two sub-problems:
  - the first was to identify the shortest path
  - the second was to calculate the length of that path
- Correctly identifying the path and not knowing how to calculate its length is better than being able to use Pythagoras' theorem but using it to calculate the length of the wrong path (Solution A) (an error of the third kind).
  - You can look up in a geometry book (or ask a friend) how to calculate the length, but you need to have identified the correct path first
- Now you could write a program to calculate the length of the path for any sized room

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

Mathematics

- A lot of the really tricky theoretical computer science problems require good skills in mathematics
- But for a great many general programming tasks nothing more than a bit of high school arithmetic/ algebra is needed
- You can become proficient in general programming without knowing advanced mathematics
- The next problem requires only basic algebra

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

ACTIVITY

The hard drive in Nick's computer has twice the capacity of Alf's. Between them their computers have 240 gigabytes of hard-drive storage. What is the capacity of each hard drive?

Verbal reasoning will not help you much here and neither will a diagram

©2008, PAUL VICKERS

Alf & Nick's hard drive

- Let x stand for Nick's hard drive and y for Alf's. We can state the following:

$$(1) \quad x + y = 240$$
$$(2) \quad x = 2y$$

- That is, in (1) we know that the size of both drives together is 240 GB and in (2) that drive x is twice the size of drive y. As x is the same value as 2y, then we can substitute this term for x in the first equation to give:

$$(3)\ 2y + y = 240, \text{ or } 3y = 240$$
$$(4)\ y = \frac{240}{3} = 80$$

- Alf's = 80 GB so Nick's = 160 GB

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

Physical models

- Physical models give you a view that natural language cannot
- They give you a clear idea of the boundaries of the thing to be built
- Physical models provide an all-round three-dimensional view which blueprints cannot
- Blueprints, on the other hand, can be rolled up and carried easily as well as showing precise and detailed measurements

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

ACTIVITY

Make a physical model of the ant-and-sugar problem

Take the inside tray of a matchbox. Make a small hole in a bottom corner and a small notch in the opposite upper corner. Take a rubber band and cut it so that it forms a single thread. Make a knot near one end and thread it through the hole so that the knot rests against the outside of the box. Pull the band reasonably tight, put it through the notch in the upper corner and make another knot so that you now have the band forming the diagonal between the two corners. Coat the band with some ink and, holding the upper corner end of the band firmly, fold out the end of the match box flat so that the rubber band rests taught against the base of the box. Now fold the end of the box back up again. If all has gone well you should now have a trail of ink marking out the shortest walking route between the two corners.

©2008, PAUL VICKERS

Benefits of the model

- The physical model gives a very clear visualization of the problem. Depending on how you manipulate the model you can have both the three-dimensional and two-dimensional views we drew earlier. The model should put to rest any confusion between the ant's route and that taken by a fly.

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

Solution description

- Once the problem has been described and understood we must solve it and then write down our solution
- The notations we used to describe the problems were to help us understand them
- As we work towards writing programs we need a language for writing our solutions which can later be translated into a real computer programming language
- Our solution language must help us to write clear and unambiguous solutions

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

ACTIVITY

Try explaining to someone how to get to your house or the post office from wherever you are right now.

Did you do well? Did they look confused? How do you capture and set out all the rules of your decision making? How many of your rules are ambiguous or incompletely stated?

©2008, PAUL VICKERS

Ambiguity

- Natural language can be ambiguous (ask a lawyer), and one idea can be expressed in multiple ways:
  - "empty the bin when it is full"
  - "if the bin is full then empty it"
  - "should the bin run out of room then you must empty it"
- Programmers must express themselves concisely and clearly:
  - computers will not interpret what we say or try to infer what we mean
  - we must tell them precisely what we want them to do
  - computers will follow your instructions to the letter **even if the instructions are wrong**

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

Pseudo-code: a language for description

- We will use a pseudo-code (aka Structured English, or Program Design Language) for writing solutions to problems
- The pseudo-code has some formal rules which make it unambiguous
- It is half-way between loose natural language and the very logically precise real computer programming languages
- It starts out quite informal in the early chapters but tightens up as programming concepts are introduced through the book
  - Many programmers use pseudo-code to express their ideas in a structured way without the complexities of a real programming language

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

Pseudo-code example

- Here is an example of a pseudo-code description of the process of getting up in the morning:

Task number

Task, or action

Semi-colon denotes end of action

```
1. Switch off alarm ;
2. Get out of bed ;
3. Wash/shower face, brush teeth, etc. ;
4. Get dressed ;
```

- By using the semi-colon as a separator we can easily see that there are four distinct actions. By numbering and putting each action on a separate line the sequence of actions is also very clear

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

Importance of sequence

- The idea of **sequence** (the order in which actions are carried out) is a vitally important programming concept
  - Things must be done in the right order to get the right results
  - e.g., a cake must not be put in the oven before the ingredients have been mixed
- Sometimes the ordering of some tasks is unimportant: does it matter whether I put my left shoe on before my right shoe?

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

ACTIVITY

Thinking of getting up in the morning, is it vital that we switch off the alarm before getting out of bed? Could we just as easily swap the ordering of the first two actions. In what circumstances might swapping the order of the actions not be possible?

©2008, PAUL VICKERS

Swapping the order

- What if the alarm clock is on the other side of the room from our bed and we cannot reach it without getting up? In this case it is very important which way round we write the instructions as they could not be followed otherwise
- Knowing the correct sequence of actions requires us not just to understand the problem of getting up in the morning, but also to know something of the **context** in which this problem is situated
- We have to know something of the relationships between the principal parts of the problem — we must know, for instance, where the alarm is in relation to the bed

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

Problem framing

- Looking at the problem to understand its boundaries and its context is called **framing**
  - (When you buy a picture to hang on your wall you need a frame to hold it)
- When we frame a problem we are describing, classifying, and relating it to other problems we have already met. Knowing if it is similar to a problem we have met before will bring to mind some relevant questions to ask and some techniques to help us solve it. We saw examples of this earlier when we considered description languages. Some problems were more easily visualized by drawing a diagram than by attempting to reason them out verbally

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

Ordering continued

- We can see other more obvious orderings:
  - we need to get out of bed before we can wash or shower
  - it is clearly not possible to wash or shower after we have got dressed (not without risking wetting your clothes)
- In examining the orderings of sequences we recognize that what seemed like an obvious and immutable sequence of actions is much more complex than we first thought
- Some ordering questions are resolved by adding conditions
  - If the alarm is reachable switch it off then get out of bed otherwise get out of bed first
  - We will deal with conditions in the next chapter

©2008, PAUL VICKERS
www.cengage.co.uk/vickers

ACTIVITY

Use the problem-solving strategy and apply any visual thinking, diagrams, or algebra techniques that you find helpful to understand the following problem and then write down your solution in pseudo-code form:

Using an electric filter machine (also called a percolator) make a pot of coffee and pour a cup

©2008, PAUL VICKERS

end of chapter 3

©2008, PAUL VICKERS