Problems!

Dr Kamal Bentahar

School of Engineering, Environment and Computing
Coventry University

03/10/2013

We are required to cook an egg for exactly 15 minutes, but instead of a timer, we are given two ropes which burn for exactly 1 hour each. The ropes, however, are of uneven densities – i.e., half the rope length-wise might take only 2 minutes to burn.

1. What is important? Numbers usually have a meaning behind them. The 15 minutes vs 1 hour, and 2 ropes were picked for a reason. . .

We are required to cook an egg for exactly 15 minutes, but instead of a timer, we are given two ropes which burn for exactly 1 hour each. The ropes, however, are of uneven densities – i.e., half the rope length-wise might take only 2 minutes to burn.

1. What is important? Numbers usually have a meaning behind them. The 15 minutes vs 1 hour, and 2 ropes were picked for a reason...
2. Simplify! We can easily time one hour (burn just one rope).

We are required to cook an egg for exactly 15 minutes, but instead of a timer, we are given two ropes which burn for exactly 1 hour each. The ropes, however, are of uneven densities – i.e., half the rope length-wise might take only 2 minutes to burn.

1. What is important? Numbers usually have a meaning behind them. The 15 minutes vs 1 hour, and 2 ropes were picked for a reason...
2. Simplify! We can easily time one hour (burn just one rope).
3. Now, can we time 30 minutes? That is half the time it takes to burn one rope. Can we burn the rope twice as fast? Yes! (Light the rope at both ends.)

We are required to cook an egg for exactly 15 minutes, but instead of a timer, we are given two ropes which burn for exactly 1 hour each. The ropes, however, are of uneven densities – i.e., half the rope length-wise might take only 2 minutes to burn.

1. What is important? Numbers usually have a meaning behind them. The 15 minutes vs 1 hour, and 2 ropes were picked for a reason...
2. Simplify! We can easily time one hour (burn just one rope).
3. Now, can we time 30 minutes? That is half the time it takes to burn one rope. Can we burn the rope twice as fast? Yes! (Light the rope at both ends.)
4. We have now learned: (1) We can time 30 minutes. (2) We can burn a rope that takes x minutes in just $x/2$ minutes by lighting both ends.

We are required to cook an egg for exactly 15 minutes, but instead of a timer, we are given two ropes which burn for exactly 1 hour each. The ropes, however, are of uneven densities – i.e., half the rope length-wise might take only 2 minutes to burn.

1. What is important? Numbers usually have a meaning behind them. The 15 minutes vs 1 hour, and 2 ropes were picked for a reason…
2. Simplify! We can easily time one hour (burn just one rope).
3. Now, can we time 30 minutes? That is half the time it takes to burn one rope. Can we burn the rope twice as fast? Yes! (Light the rope at both ends.)
4. We have now learned: (1) We can time 30 minutes. (2) We can burn a rope that takes x minutes in just $x/2$ minutes by lighting both ends.
5. Work backwards: if we had a rope of burn-length 30 minutes, that would let us time 15 minutes. Can we remove 30 minutes of burn-time from a rope?

We are required to cook an egg for exactly 15 minutes, but instead of a timer, we are given two ropes which burn for exactly 1 hour each. The ropes, however, are of uneven densities – i.e., half the rope length-wise might take only 2 minutes to burn.

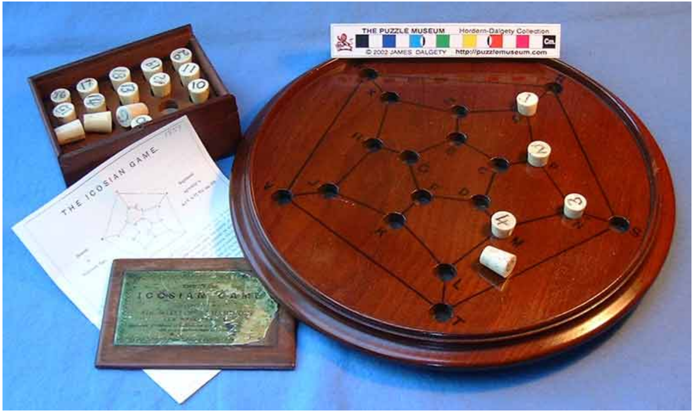
1. What is important? Numbers usually have a meaning behind them. The 15 minutes vs 1 hour, and 2 ropes were picked for a reason…
2. Simplify! We can easily time one hour (burn just one rope).
3. Now, can we time 30 minutes? That is half the time it takes to burn one rope. Can we burn the rope twice as fast? Yes! (Light the rope at both ends.)
4. We have now learned: (1) We can time 30 minutes. (2) We can burn a rope that takes x minutes in just $x/2$ minutes by lighting both ends.
5. Work backwards: if we had a rope of burn-length 30 minutes, that would let us time 15 minutes. Can we remove 30 minutes of burn-time from a rope?
6. We can remove 30 minutes of burn-time from Rope 2 by lighting Rope 1 at both ends and Rope 2 at one end.

We are required to cook an egg for exactly 15 minutes, but instead of a timer, we are given two ropes which burn for exactly 1 hour each. The ropes, however, are of uneven densities – i.e., half the rope length-wise might take only 2 minutes to burn.

1. What is important? Numbers usually have a meaning behind them. The 15 minutes vs 1 hour, and 2 ropes were picked for a reason...
2. Simplify! We can easily time one hour (burn just one rope).
3. Now, can we time 30 minutes? That is half the time it takes to burn one rope. Can we burn the rope twice as fast? Yes! (Light the rope at both ends.)
4. We have now learned: (1) We can time 30 minutes. (2) We can burn a rope that takes x minutes in just $x/2$ minutes by lighting both ends.
5. Work backwards: if we had a rope of burn-length 30 minutes, that would let us time 15 minutes. Can we remove 30 minutes of burn-time from a rope?
6. We can remove 30 minutes of burn-time from Rope 2 by lighting Rope 1 at both ends and Rope 2 at one end.

Icosian Game

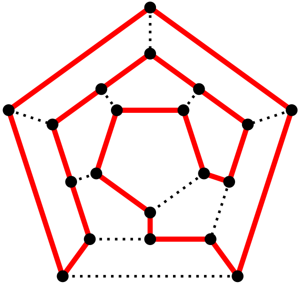
Irish mathematician William Hamilton (Dublin, 1857)



Problem (Hamiltonian Cycle)

Given a graph, decide if it contains a path that visits every node exactly once and terminates at the same starting node.

Icosian Game



Problem (Hamiltonian Cycle)

Given a graph, decide if it contains a path that visits every node exactly once and terminates at the same starting node.

Problem

Given the set $S = \{2, 3, 5, 7, 11, 13\}$, decide if there is a subset of S whose sum is 15.

Problem

Given the set $S = \{2, 3, 5, 7, 11, 13\}$, decide if there is a subset of S whose sum is 15.

Problem (Subset-Sum Problem)

Given a set $S = \{x_1, x_2, \ldots, x_n\}$ of integers, and an integer t (called target) decide if there is a subset of S whose sum is equal to t.

Problem

Given the set $S = \{2, 3, 5, 7, 11, 13\}$*, is it possible to split it into 2 sets with equal sums?*

Problem

Given the set $S = \{2, 3, 5, 7, 11, 13\}$, is it possible to split it into 2 sets with equal sums?

Problem (Partition Problem)

Given a set $S = \{x_1, x_2, \ldots, x_n\}$ of numbers, decide if it can be partitioned into two sets such that they both have the same sums.

Problem (Satisfiability)

Given an expression that consists of Boolean variables connected by the symbols $\neg, \wedge, \vee, \implies$ *, decide if there is a way of assigning the values* *true* *and* *false* *to the variables so that the expression is* *true*.

Problem (Satisfiability)

Given an expression that consists of Boolean variables connected by the symbols $\neg, \wedge, \vee, \implies$, decide if there is a way of assigning the values true and false to the variables so that the expression is true.

Problem (Clique)

A clique in a graph is a set of nodes for which any two are connected. Given a graph and an integer n, decide if it contains a clique with n nodes.

Problem (Satisfiability)

Given an expression that consists of Boolean variables connected by the symbols $\neg, \wedge, \vee, \implies$ *, decide if there is a way of assigning the values true and false to the variables so that the expression is true.*

Problem (Clique)

A clique in a graph is a set of nodes for which any two are connected. Given a graph and an integer n*, decide if it contains a clique with* n *nodes.*

Problem (A Diophantine quadratic equation in two variables)

Given three positive integers a, b, c*, decide if the equation*

$$ax^2 + by = c$$

has a solution in positive integers.

Types of problems

- Decision
- Search
- Computation/Construction
- Counting
- Optimization
- ...

How hard is a problem?

Here are some example problems – how hard are they to solve?

- What is $1 + 1$?

How hard is a problem?

Here are some example problems – how hard are they to solve?

- What is $1 + 1$?
- What is the shortest route across the rail network from Coventry to London?

How hard is a problem?

Here are some example problems – how hard are they to solve?

- What is $1 + 1$?
- What is the shortest route across the rail network from Coventry to London?
- What is the shortest tour around all the universities in the UK and back to your starting point (by car say)?

How hard is a problem?

Here are some example problems – how hard are they to solve?

- What is $1 + 1$?
- What is the shortest route across the rail network from Coventry to London?
- What is the shortest tour around all the universities in the UK and back to your starting point (by car say)?

How hard is a problem?

Here are some example problems – how hard are they to solve?

- What is $1 + 1$?
- What is the shortest route across the rail network from Coventry to London?
- What is the shortest tour around all the universities in the UK and back to your starting point (by car say)?

Can you say why you feel that the first one is "easier" than the other two?

How hard is a problem?

Here are some example problems – how hard are they to solve?

- What is $1 + 1$?
- What is the shortest route across the rail network from Coventry to London?
- What is the shortest tour around all the universities in the UK and back to your starting point (by car say)?

Can you say why you feel that the first one is “easier” than the other two? What about the last two? Is one much harder than the other, or are they both about the same?

How can we measure the hardness of a problem?

- By the type/sophistication of the machine/process required to solve it?

How can we measure the hardness of a problem?

- By the type/sophistication of the machine/process required to solve it?
  - Real physical machines

How can we measure the hardness of a problem?

- By the type/sophistication of the machine/process required to solve it?
  - Real physical machines
  - Theoretical (imaginary) machines

How can we measure the hardness of a problem?

- By the type/sophistication of the machine/process required to solve it?
  - Real physical machines
  - Theoretical (imaginary) machines
- By the amount of resources used by the machine?

How can we measure the hardness of a problem?

- By the type/sophistication of the machine/process required to solve it?
  - Real physical machines
  - Theoretical (imaginary) machines
- By the amount of resources used by the machine?
  - Processor time

How can we measure the hardness of a problem?

- By the type/sophistication of the machine/process required to solve it?
  - Real physical machines
  - Theoretical (imaginary) machines
- By the amount of resources used by the machine?
  - Processor time
  - Memory space

How can we measure the hardness of a problem?

- By the type/sophistication of the machine/process required to solve it?
  - Real physical machines
  - Theoretical (imaginary) machines
- By the amount of resources used by the machine?
  - Processor time
  - Memory space
- By the level of difficulty encountered by the (human) solver of the problem?

Problems and problem instances

The 3 examples stated earlier were actually not *problems* but **instances of problems.**

Problems and problem instances

Problems: Generalization of a problem instance.
Not useful to do just $1 + 1$ or $1 + 2$.

Problems and problem instances

For a specific problem instance, we could measure exactly the amount of processor time and memory capacity required to solve it, using some suitable process.
However, when solving a general problem, we cannot always say exactly what resources will be used.

Problems and problem instances

- We normally express resource usage as a function of the **problem's size**.
- Also, when we ask questions about whether a problem is solvable by some machine, it is normal to allow the machine to have unlimited memory capacity and unlimited time as all problems become unsolvable at some point if finite limits are in place.
- It is for this reason, amongst others, that theoretical machines are used in classifying hardness.

O-notation scale

- Polynomial: $1, n, n^2, n^3, \ldots$ (Also, $n^k (\log n)^\ell$)
- Exponential: $2^n, 3^n, \ldots$
- Combinatorial: $n!, n^n, \ldots$

Problems and Processes

Difficulty: the type of machine, time and space required may depend on our choice of process used to solve the problem.

Problems and Processes

Difficulty: the type of machine, time and space required may depend on our choice of process used to solve the problem.

Problem (Addition of integers)

Find $c = a + b$.

Suggested solution:

1: Select a random number r
2: **if** $r - b = a$ **then**
3: $\quad c \leftarrow$ and stop
4: **else**
5: $\quad$ repeat
6: **end if**

Problems and Processes

When we measure a problem's hardness in terms of resources or machines, we are really measuring the hardness of a process used to solve the problem. The hardness of the problem should be taken to be the hardness of the most efficient process capable of solving it.

Problems and Processes

Interestingly, there exist some problems for which the most efficient processes known or even possible are “guess and check” methods – and something very interesting happens when the number of possible guesses is infinite. . . !

A needle in a haystack

Problem:

Given any (finite) haystack H, decide whether H contains a needle.

A needle in a haystack

Problem:

Given any (finite) haystack H, decide whether H contains a needle.

This problem is easy, though perhaps a little tedious, to solve: simply search every location within the haystack in some predefined order and terminate with the answer yes should you come across a needle. If you complete the search and no needle has been found, terminate with the answer no.

A needle in a haystack

Problem:

Given any (finite) haystack H, decide whether H contains a needle.

This problem is a type of **decision problem**: given some data (the haystack) decide if the data has a certain property (needle containment).

A needle in a haystack

We may divide all possible instances of the problem into yes instances (haystacks with needles) and no instances (haystacks without needles) using our process.

- What happens if the haystacks are infinite?
- Can you still divide them into yes and no haystacks?
- How else could we divide them up?

A needle in a haystack

We may divide all possible instances of the problem into yes instances (haystacks with needles) and no instances (haystacks without needles) using our process.

- What happens if the haystacks are infinite?
- Can you still divide them into yes and no haystacks?
- How else could we divide them up?

Complexity Onion