3 Nervous Couples

Three couples (president and bodyguard) wish to cross a river. They have one boat, that can carry at most two people, making several trips across the river necessary. The current is swift, the river is deep, and none of them can swim.

Each president is nervous about the other presidents' bodyguards, so no president is willing to be on a bank or in the boat with another president's bodyguard, unless his or her own bodyguard is also present.

How can all three couples cross the river?

Decomposition

We could try to break the problem down into stages:

- get one couple across, then another couple across, and then the last couple across
- get all the bodyguards across, then all the presidents
- get all the presidents across, then all the bodyguards

The first approach won't work at all… why?

The second approach looks hard, because if we send one bodyguard across the river then his president would be on the bank with the other bodyguards, which isn't allowed.

The third approach looks easier (to start), since we can send a president across without his bodyguard, as long as we leave all the bodyguards where they are.

Symmetry

We could look for symmetries between, or with respect to, the entities that make up the problem:

- presidents and bodyguards are not symmetrical; they operate under different constraints
- in a sense, the two banks of the river are symmetrical (call them left and right, and suppose everyone starts on the left bank); any movement from left to right is reversible by a symmetric movement

The third approach looks easier (to start), since we can send a president across without his bodyguard, as long as we leave all the bodyguards where they are.

Representation

We need some compact, precise way to represent things:

- the current state of the world (who's on the left bank, who's on the right bank, who's on the river)
- the actions that are being performed (who's travelling and in which direction)

We don't need to distinguish the three presidents from each other (or the three bodyguards) as long as we can keep track of president/bodyguard couples.

It's important to NOT label things that don't need to be labeled --- unnecessary labels introduce unnecessary distinctions (information) into the problem and make the analysis more complicated.

Representation

We will represent a president by `P` and a bodyguard by `B` and a couple by `C`.

We will represent the *current state* of the world this way:

```
{who's on left bank |river| who's on right bank}
```

For example, the starting state would be represented as:

```
{ 3C || }
```

This indicates three couples on the left bank, and no one on the right bank.

A later state might be represented as:

```
{ 2C,1B || 1P }
```

This indicates two couples and one bodyguard (belonging to a different president) on the left bank, and one president (whose bodyguard is not present) on the right bank.

Representation

We will represent a transition between stages this way:

```
who's on left bank | who's in the boat | who's on right bank
```

For example, from the starting state

```
{ 3C || }
```

we could make the transition (take the action if you prefer) represented by

```
>   1C,2B | 2P |
```

In words, two presidents are travelling (from left to right), leaving their two bodyguards and the other couple on the left bank.

This action will result in a new state:

```
{ 1C,2B || 2P }
```

Defining Goals

Clearly, we want to go from the starting state

```
{  3C  ||  }
```

to the final state

```
{  ||  3C  }
```

Let's try assuming that the solution is symmetrical; that is, whatever sequence of actions we take to go from the starting state to the goal state is exactly reversible (it is).

Let's also set an intermediate goal to achieve the following state:

```
{  3B  ||  3P  }
```

(Of course, this might not be a good choice, but we need to start with some idea in mind.)

First Steps

```
     {3C  ||  }
>    1C,2B  |2P|
     {1C,2B  ||  2P}
<    1C,2B  |1P|  1P
     {2C,1B  ||  1P}
>    3B  |2P|  1P
     {3B  ||  3P}
     .  .  .
```

The notation we've defined makes it easy to keep track of the current situation, and how we reached it.

End Game

Symmetry shows us one way that the final goal might be reached:

```
      . . .
      {3P || 3B}
>     1P |2P| 3B
      {1P || 2C,1B}
<     1P |1P| 1C,2B
      {2P || 1C,2B}
>     |2P| 1C,2B
      { || 3C}
```

But… not from the intermediate state

```
{ 3B || 3P }
```

Middle Crossings

```
{3B || 3P}
```

Now, we need to discover whether we can go from our first intermediate state

```
1C |1C| 1C
```

This is the only purely symmetrical action… perhaps it's a good fit for the middle of this transition…

```
{3P || 3B}
```

… to the state from which we can reach our final goal

Middle Crossings

```
     {3B || 3P}
                                 <     3B |1P| 2P
                                       {1C,2B || 2P}
                                 >     1C |2B| 2P
                                       {1C || 2C}
<    1C |1C| 1C
                                       {2C || 1C}
                                 >     2P |2B| 1C
                                       {2P || 1C,2B}
                                 <     2P |1P| 3B
     {3P || 3B}
```

Full Solution

```
    {3C  ||  }
>   1C,2B |2P|
    {1C,2B || 2P}
<   1C,2B |1P| 1P
    {2C,1B || 1P}
>   3B |2P| 1P
    {3B || 3P}
<   3B |1P| 2P
    {1C,2B || 2P}
>   1C |2B| 2P
    {1C || 2C}
<   1C |1C| 1C
    {2C || 1C}
>   2P |2B| 1C
    {2P || 1C,2B}
<   2P |1P| 3B
    {3P || 3B}
>   1P |2P| 3B
    {1P || 2C,1B}
<   1P |1P| 1C,2B
    {2P || 1C,2B}
>   |2P| 1C,2B
    {  || 3C}
```

Hoare

Charles Antony Richard Hoare

- invented Quicksort
- ALGOL 60
- Communicating Sequential Processes
- Hoare logic
- axiomatic specification of PLs

1980	ACM Turing Award
1982	FRS
2000	knighted
2005	FREng
2011	IEEE John von Neumann Medal