The PicoBot Challenge!

Adapted from the "Nifty Assignment" presented at SIGCSE 2010 by Zach Dodds and Wynn Vonnegut, Harvey Mudd College.

Computation - "A process that transitions from state to state under control of a program."

The key elements in our definition of a computation are:

- Process
- State
- Program

Process implies an algorithm, and a program defines the algorithm in terms of actions and state transitions. State defines the internal context of the computation. It is the state of the process and should not be confused with the state of the external view of the process.

In this lab, we are going to define computations for a robot called a PicoBot. "YAK! – Yet Another Karel!" you say. Actually, it is quite different.

A PicoBot is aware of its immediate surroundings in all four cardinal directions. That is, it knows whether or not it can move to the north, east, west or south. A PicoBot maintains a single integer variable which denotes its current state. A PicoBot program consists of a series of moves depending on the surroundings and the state. The resulting process can solve a number of interesting problems.

At each step, the PicoBot can optionally make a move according to its surroundings and it will always specify its next state. A state in a PicoBot program must specify exactly what the PicoBot is to do for each of the possible surroundings.

For example, a program that causes a PicoBot to move north until it can no longer move might be specified as follows:

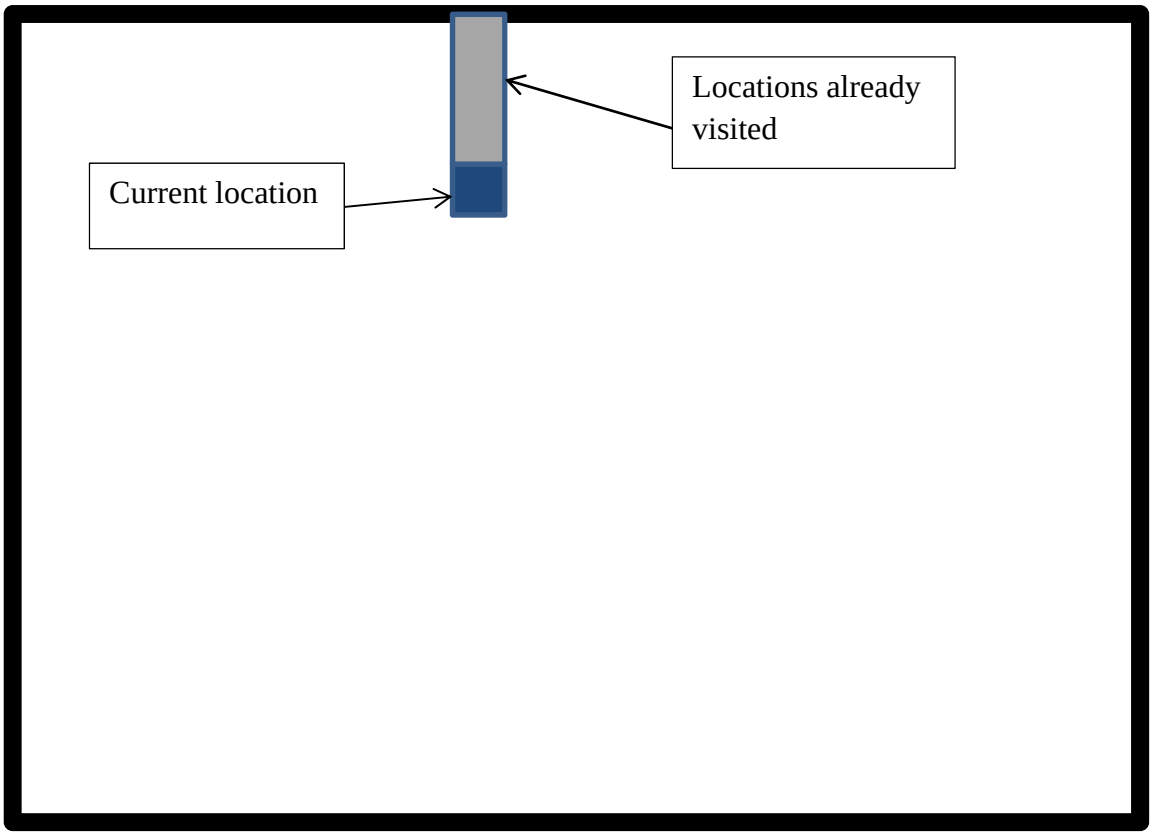
```
State 0:
     If North is not blocked then move North and set state to 0
     else set state to 1
State 1:
     No matter what the surroundings are, set state to 1
```

Notice that each statement specifies what to do for all possible values of the surroundings.

Challenge Number 1 – The Empty Room

In this challenge, the PicoBot starts somewhere inside of an enclosed room that is otherwise empty. Your challenge is to specify a PicoBot program that will cause the PicoBot to visit every location within the room. This is equivalent to the "Roomba" vacuum robot problem.

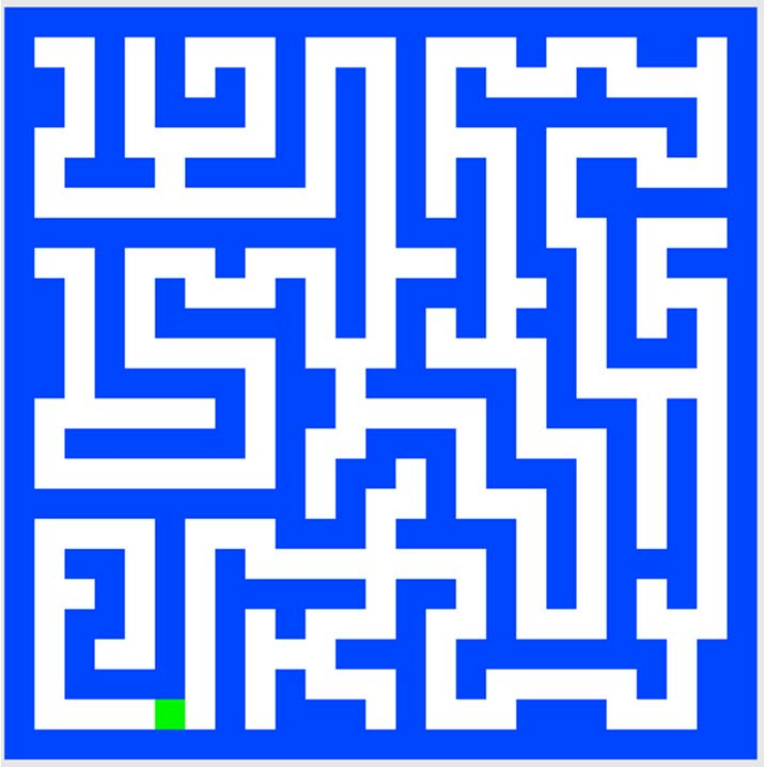
Is it possible to devise a room where your PicoBot is incapable of visiting every possible location? In other words, are there rooms that are not computable using your PicoBot?



Example room showing PicoBot and area already visited

Challenge Number 2 – The Connected Maze

In this challenge, the PicoBot will start somewhere within the simply connected maze and your job is to write a program that causes the PicoBot to visit every location within the maze. An example is shown below.



Instructions for completing the challenge

1. You may work in groups of 1 or 2.
2. Using appropriate pseudocode as shown in the introduction, you must define a sequence of commands that cause the PicoBot to perform its task. Your program is organized as a series of states. Each state has a sequence of commands which taken together specify a move for all possible surroundings and next state values. The example given in the introduction shows how to use if and else statements to efficiently accomplish this. Remember that a PicoBot 'move' may cause the PicoBot to remain in its current location.
3. The deliverable for this lab is a fully documented PicoBot pseudocode program. You may begin lines with the character # to denote a comment that is not part of the PicoBot program. Commenting is expected.
4. Your work should be captured in your class notebook. Each student is responsible for having the program design and implementation in their notebook.
5. Each group will deliver a document, printed out, containing their fully commented code for each challenge. The names of all students in the group must be on this document.