

PUT YOUR NAME HERE:

1. Imagine a checked board. There is a robot on the board that can go one square up or right at a time. Some squares are blocked by stones. Using pseudocode write a (simple) algorithm for the robot to make moves while it can. Don't forget that board has bounds (so stones are not the only reason why robot might not be able to make a certain move).
2. Suppose the board from question 1 is  $m$  squares wide and  $n$  squares long how many moves can the robot from question 1 make at most?
3. Consider two algorithms:

```
procedure  $F(n)$   
  if ( $n > 0$ ) then  
    result is  $n + F(n - 1)$   
  else  
    result is 0
```

```
procedure  $G(n)$   
   $A \leftarrow 0$   
  while ( $n > 0$ ) do  
    ( $A \leftarrow A + n$   
      $n \leftarrow n - 1$ )  
  result is  $A$ 
```

- (a) What is the result of  $F(5)$ ? Show intermediate steps of this computation. How many times **if** statement was executed?
- (b) What is the result of  $G(6)$ ? Show intermediate steps of this computation. How many times  $A$  was assigned some value?
- (c) What  $F(n)$  actually computes?
- (d) What  $G(n)$  actually computes?
- (e) Do you think results of  $F(n)$  and  $G(n)$  may be different for some  $n$  or they are always the same? Explain.