Programming Methodology

Lab 11: Program Correctness

Wednesday, April 24th, 2024

Weekly Lab Agenda

- Go over reminders/goals
- Review past material
- Work in groups of 2-3 to solve a few exercises
    - Lab leaders will assign new groups this week
- Discussion leaders will walk around and answer questions
- Solutions to exercises will be reviewed as a class
- Attendance taken at the end

Reminders

- You should be working on your group project.
- Have a planned timeline to avoid last minute rush!
- Reach out to us if there is any issues in your team.
  - Private post on campuswire or
  - Email to mkuechen@umass.edu

Today's Goals

- Practice working on program correctness.

Question 1

Please write your solution to question 1 on paper. You will submit your work to gradescope before we review the solution, and we'll grade your work to give you feedback.

Question 1

What is the largest value A for which the loop below terminates with n=20?

Use an invariant to justify your answer.

#Fall 2022 Final Exam

```
let n: number = 0;

let s: number = A;

while (s <= 100) {

        if (n % 2 > 0) {

                s += n;

        } else {

        }

        n = n + 1;

}
```

Question 1

Please submit to gradescope now before we go over the solution. We'll grade these to give you feedback, but the score you receive will not impact your grade. Your grade for this lab is as always based on attendance only.

Question 2

In class yesterday we saw the following code, which finds the index of the biggest element of array arr between indexes a and b, inclusive.

```
function biggest(arr, a, b) {
  let big = a;
  for (let i=a+1; i < b; i=i+1) {
    // big: the index of the biggest element in arr[a..i-1]
    if (arr[i] > arr[big]) {
      big = i;
      // big is the index of the biggest element in arr[a..i]
    }
    else {
      // big is the index of the biggest element in arr[a..i-1]
      // and arr[i] is not bigger than arr[big]
      // therefore big is the index of the biggest element in arr[a..i]
    }
    // "implied i = i+1"
    // big is the index of the biggest element in arr[a..i-1]
  }
  return big;
}
```

Some assertions are missing from the code listed here, but we saw that the invariant for variable big is maintained.

Question 2

In class yesterday we saw the following code, which finds the index of the biggest element of array arr between indexes a and b, inclusive.

```
function biggest(arr, a, b) {
  let big = a;
  for (let i=a+1; i < b; i=i+1) {
    // big: the index of the biggest element in arr[a..i-1]
    if (arr[i] > arr[big]) {
      big = i;
      // big is the index of the biggest element in arr[a..i]
    }
    else {
      // big is the index of the biggest element in arr[a..i-1]
      // and arr[i] is not bigger than arr[big]
      // therefore big is the index of the biggest element in arr[a..i]
    }
    // "implied i = i+1"
    // big is the index of the biggest element in arr[a..i-1]
  }
  return big;
}
```

Let's take a look at the stopping criteria of this loop, to see if it's the right one.

Question 2

In class yesterday we saw the following code, which finds the index of the biggest element of array arr between indexes a and b, inclusive.

```
function biggest(arr, a, b) {
  let big = a;
  for (let i=a+1; i < b; i=i+1) {
    // big: the index of the biggest element in arr[a..i-1]
    if (arr[i] > arr[big]) {
      big = i;
      // big is the index of the biggest element in arr[a..i]
    }
    else {
      // big is the index of the biggest element in arr[a..i-1]
      // and arr[i] is not bigger than arr[big]
      // therefore big is the index of the biggest element in arr[a..i]
    }
    // "implied i = i+1"
    // big is the index of the biggest element in arr[a..i-1]
  }
  return big;
}
```

If big is the index of the biggest element in arr[a..i-1], what value do we want i-1 to have when we exit the loop?