

Assignment 3 – More Recursion and Queues

Write pseudo-code not Java for problems requiring code. You are responsible for the appropriate level of detail.

Q1 and Q2 are intended to help you get comfortable with recursion by thinking about something familiar in a recursive manner. Q3 – Q4 are practice in working with non-trivial recursive functions. Q5 and Q6 deal with the idea of conversion between iteration and recursion.

1. Write a recursive algorithm to compute $a+b$, where a and b are nonnegative integers.
2. Let A be an array of integers. Write a recursive algorithm to compute the average of the elements of the array. Solutions calculating the sum recursively, instead of the average, are worth fewer points.
3. A generalized Fibonacci function is like the standard Fibonacci function,, except that the starting points are passed in as parameters. Define the generalized Fibonacci sequence of f_0 and f_1 as the sequence $\text{gfib}(f_0, f_1, 0), \text{gfib}(f_0, f_1, 1), \text{gfib}(f_0, f_1, 2), \dots$, where
$$\begin{aligned}\text{gfib}(f_0, f_1, 0) &= f_0 \\ \text{gfib}(f_0, f_1, 1) &= f_1 \\ \text{gfib}(f_0, f_1, n) &= \text{gfib}(f_0, f_1, n-1) + \text{gfib}(f_0, f_1, n-2) \text{ if } n > 1\end{aligned}$$

Write a recursive method to compute $\text{gfib}(f_0, f_1, n)$.

4. Ackerman's function is defined recursively on the nonnegative integers as follows:

$$\begin{aligned}a(m, n) &= n + 1 && \text{if } m = 0 \\ a(m, n) &= a(m-1, 1) && \text{if } m \neq 0, n = 0 \\ a(m, n) &= a(m-1, a(m, n-1)) && \text{if } m = 0, n \neq 0\end{aligned}$$

Using the above definition, show that $a(2,2)$ equals 7.

5. Convert the following recursive program scheme into an iterative version that does not use a stack. $f(n)$ is a method that returns TRUE or FALSE based on the value of n , and $g(n)$ is a method that returns a value of the same type as n (without modifying n).

```
int rec(int n)
{
    if ( f(n) == FALSE ) {
        /* any group of statements that do not change the value of n */
        return (rec(g(n)));
    } //end if
    return (0);
} //end rec
```

6. Develop an ADT specification for a priority queue. A priority queue is like a FIFO queue except that items are ordered by some priority setting instead of time. In fact, you may think of a FIFO queue as a priority queue in which the time stamp is used to define priority.