University of Nevada, Las Vegas Computer Science 477/677 Fall 2024

Answers to Assignment 2: Due Friday September 13, 2024

Name:____

You are permitted to work in groups, get help from others, read books, and use the internet.
To turn in the homework, follow instructions given by the graduate assistant, Sepideh Farivar. at `farivar@unlv.nevada.edu`.

1. In an application, items of a priority queue represent **unfulfilled obligations**.
2. **pop** and **push** are operators of the ADT **stack**
3. **enqueue** and **dequeue** are operatiors of the ADT **queue**
4. **fetch** and **store** are operatiors of the ADT **array**
5. What is the asymptotic time complexity, in terms of n, of this code fragment?

```
for(int i = 1; i < n; i = 2*i)
 cout << "Hello, world!" << endl;
```

$\Theta(\log n)$

6. What is the asymptotic time complexity, in terms of n, of this code fragment?

```
for(int i = 1; i < n; i++)
 for(int j = i; j < n; j = 2*j)
  cout << "Hello, world!" << endl;
```

Substitute $t = \log j$. The inner loop is now

```
 for(int t = log i; t < log n; t++)
```

which iterates exactly $\log n - \log i$ times.[1] The time complexity is estimated by the integral

$$\int_{x=1}^{n} (\ln n - \ln x)dx = (x\ln n - x\ln x + x)\Big|_{x=1}^{x=n} = n\ln n - n\ln n + n - \ln n - 1 = \Theta(n)$$

7. What is the asymptotic time complexity, in terms of n, of this code fragment?

What's missing in this problem is the initial value of m. It turns out the the asymptotic answer is the same, regardless of that value. But we will assume that initially m = 0.

```
for(int i = 0; m < n; i++)
 m = m+i;
```

[1] For any constants A and B, the loop `for(int t = A; t < B; t++)` iterates B − A times.

If the loop iterates k times, the time complexity is $\Theta(k)$. If $m = 0$ initially, then at the end of the loop $m = 1 + 2 + \cdots k = \frac{k(k+1)}{2} = \Theta(k^2)$. By the loop condition, $m = \Theta(n)$ at that time. Thus $k^2 = \Theta(n)$, hence the time complexity $k = \Theta(\sqrt{n})$.

8. The following code sorts an array A[N]. What is the algorithm used? the possible answers are bubblesort, selection sort, and insertion sort. Assume that “swap(x,y)” exchanges the values of x and y.

```
for(int i = 0; i < N; i++)
 for(int j = i+1; j < N; j++)
  if(A[j] < A[i]) swap(A[i],A[j]);
```

selection sort

9. The C++ code below implements a function, “mystery.” What does it compute? Assume that n is positive.

```
 float mystery(float x, int n)
  {
   float y = x;
   float z = 1.0;
   int m = n;
   while(m > 0)
    {
     if(m%2) // that means m is odd
      z = z*y;
     y = y*y;
     m = m/2;
    }
   return z;
  }
```

x^n

10. The following C++ code takes as input a non-negative number n.

```
int n;
cout << "Enter a non-negative integer." << endl;
cin >> n;
assert(n >= 0);
int m = 0;
// loop invariant holds here
while(n > 0)
 {
   // loop invariant holds here
  m = m+2;
   // loop invariant does NOT hold here
  n = n-1;
   // loop invariant holds here
 }
   // loop invariant holds here
cout << "The answer is " << m << endl;
```

(a) What is the purpose of the code? It outputs twice the input value of n..

(b) What is the loop invariant of the loop? $m + 2n =$ Twice the input value of n