

## CSC 340 HOMEWORK 1

NAME: \_\_\_\_\_

You must show work that justifies your answers for each of the following problems. Submit by email when done. This homework is due Friday at the end of Week 3.

---

1. Consider the following algorithmic problem:

**Input:** An array  $a[0..N-1]$  of integer and an integer  $X$ .

**Output:** The least integer  $k$  such that  $a[k] = X$ , if such a  $k$  exists, or -1 otherwise.

Consider the following algorithm for solving this problem.

```
int k = 0;
bool found = false;
// 0.
while (k < N)
{
    // 1.
    if (a[k] == X)
    {
        // 2.
        found = true;
        // 3.
        break;
    }
    k = k + 1;
    // 4.
}
// 5.
```

Answer the following questions about this algorithm:

- (1) What is the strongest statement (invariant) involving the variables  $k$  and  $found$  that you can say holds at the point labeled 0?
- (2) What is the strongest statement (invariant) involving the variables  $k$  and  $found$  that you can say holds at the point labeled 1?
- (3) What is the strongest statement (invariant) involving the variables  $k$ ,  $found$ , and elements of the array  $a[ ]$  that you can say holds at the point labeled 2?
- (4) What is the strongest statement (invariant) involving the variables  $k$ ,  $found$ , and elements of the array  $a[ ]$  that you can say holds at the point labeled 3?
- (5) What is the strongest statement (invariant) involving the variables  $k$ ,  $found$ , and elements of the array  $a[ ]$  that you can say holds at the point labeled 4?

- (6) What is the strongest statement (invariant) involving the variables `k`, `found`, and elements of the array `a[ ]` that you can say holds at the point labeled 5?

**2.** Consider the following program fragment, where we assume that `N` is a positive integer.

```
int k = 0;
while (k < N)
{
    if (k % 2 == 0)
    {
        k --;
    }
    else
    {
        k = k + 2;
    }
}
```

Prove that the loop in this program fragment will eventually terminate, or give an example of an input value for `N` that will make the program run forever.

**3.** Consider the following program fragment, where we assume that `N` is a positive integer.

```
int k = 0;
while (k != N)
{
    if (k % 2 == 0)
    {
        k --;
    }
    else
    {
        k = k + 2;
    }
}
```

Prove that the loop in this program fragment will eventually terminate, or give an example of an input value for `N` that will make the program run forever.

**4.** Give two different pairs of values for  $(N, K)$ , where  $N$  is a positive integer and  $K$  is a positive real number, such that

$$n^3 + 70n^2 + 100 \leq K(n^3 + 10n)$$

for all  $n \geq N$ .

**5.** Prove that  $n^3 \in \Omega(100n^2 + 500n + 10)$ .

**6.** A certain algorithm performs  $n$  iterations while solving a problem instance of size  $n$ . Each such iteration requires  $n^2$  basic operations.

- (1) How many basic operations does the algorithm perform?
- (2) Can you tell for certain that the problem is the complexity class  $O(n^4 + n^2)$ ?
- (3) Can you tell for certain that the problem is the complexity class  $\Omega(n^2)$ ?

7. A certain algorithm performs  $n$  iterations while solving a problem instance of size  $n$ . For each  $k = 1, \dots, n$ , the algorithm performs  $2k - 1$  basic steps. What is the total number of basic steps performed by the algorithm?
8. Is the problem of determining whether an array of strings is sorted, where the strings have a maximum length of 100 characters NP-Complete? Justify your answer.