



Closed book/closed notes. Show all of your work clearly in the space provided or on the additional page at the end of the exam. If the additional page is used, be sure to clearly label the content for each problem. Be sure to *read each problem carefully*. You should answer all six questions. Note: The exam is double-sided.

**1.** (15 points) Agree or disagree with the following statement. Be sure to explain your reasoning.

Passing an object by value instead of by reference may have a significant effect on the asymptotic time complexity of the algorithm.



2. (10 points) Describe an application where a **vector** would be preferred over a **list**. Be sure to explain your reasoning.

3. (10 points) Describe an application where a **list** would be preferred over a **vector**. Be sure to explain your reasoning.

4. (15 points) Consider the following function.

```
#include <vector>
2 #include <cassert>
using std::vector;

4
double findKthLargest(const vector<double>& nums, unsigned int k)
6 {
    unsigned int N = nums.size();
8     assert(N>0);
    double largest = nums[0];
10    unsigned int index = 0;
    for(unsigned int i=0; i<k; ++i) {
12        for(unsigned int j=1; j<N-i; ++j) {
            if(nums[j]>largest) {
14                largest = nums[j];
                    index = j;
16            }
        }
18        swap(nums[N-i], nums[index]);
    }
20    return largest;
}
```

Assume that `swap(double& x, double& y)` is an  $O(1)$  function that swaps the values found in  $x$  and  $y$ . Using big-oh notation, describe the overall worst case time complexity for the function. Be sure to explain your reasoning and state any additional assumptions that you make. Hint: Use  $N$  and  $k$  to characterize the input size.



**5. (a)** (15 points) Write a recursive function, `power`, which accepts a `double` and an `unsigned int` as input and returns the value of the `double` raised to the `unsigned int` power.

**(b)** (10 points) Using big-oh notation, describe the overall worst case time complexity for your algorithm. Be sure to explain your reasoning.



6. (a) (15 points) Write the generic algorithm

`replace(itr start, itr stop, targetValue, replaceValue)`

that replaces all occurrences of `targetValue` between the `start` and `stop` iterators with `replaceValue`. Your algorithm should be generic in the sense that it should work with any container class containing objects from any class for which a default constructor, equality (`==`) operator, and assignment (`=`) operator have been defined. The algorithm should return `false` only if no replacements were made.

(b) (10 points) Using big-oh notation, describe the overall worst case time complexity for your algorithm. Be sure to explain your reasoning.



Additional work area for any problem. Clearly identify to which problem the work on this page is related.