CS211 – Data Structures

Time: 90 Minutes | Midterm Exam – November 22 | Total Marks: 50

Q1: Algorithm **A** does a particular task in $5n^2$ steps. Algorithm **B** does the same task in $25n+1000$ steps: {1+1+1+5=8}

a. *What is the best case time complexity of each algorithm?*

b. *What is the worst case time complexity of each algorithm?*

c. *Which algorithm is more efficient by Big-O standards? Why?*

d. *Under what conditions, if any, would the "less efficient" algorithm execute more quickly than the "more efficient" algorithm?*

Q2: Assume a queue ADT implemented using arrays with rear_index and front_index initialized with a value of 3. How the internal array of the queue with size 8 would look like after inserting the following values? {7}

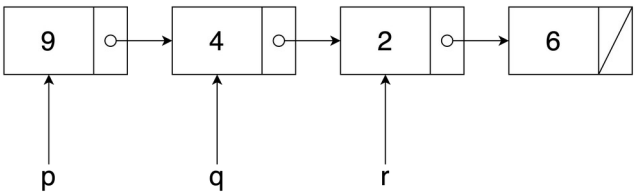
2,3,7,4,6,5,1

Q3: What would be the output of the following code? {5}

```
queue<int> q;
stack<int> s;

for (int i=1;i<=5;i++) {
  q.push(i);
  s.push(i*2);
}

while (!s.empty()) {
  q.push(s.top());
  s.pop();
}

while (!q.empty()) {
  cout<<s.top()<<" ";
  q.pop();
}
```

Q4: Convert the following expression into postfix: 5+3-7*4/2-1 {10}

Q5: Given the following linked structures, what would be result of the following expressions? In case of an error, explain it. {10}



a. `p->data->next`

b. `q->next == r`

c. `q->prev == p`

d. `p->next->next->data`

e. `p->data + r->data`

f. `q->next + r->next`

g. `r->next->next->data`

h. `r->next == 6`

i. `(p+sizeof(node)) == q`

j. `p->next->next == r`

Q6: Write code for displaying an STL based list passed as constant and by references {10}

```
template<typename T>
void display(const list<T> &l) {
```