## Exercise 4

1. Give example c++ code and test data where a program error is detected through path coverage criterion, and not necessarily through statement coverage criterion.

2. Give example c++ code for a function that may throw exceptions of type *int* or *char* only, has a nested *try* block, and contains a *catch* block with an ellipsis.

3. Give example c++ code that uses an associative STL container, a constant iterator, and a generic function.

## Exercise 5

1. What is printed by the program below?

```
void function( int n, int m ){
  cout << "range " << n << " " << m << endl;
  if( n < m ) {
    ++n;
    function( n, m );
    --m;
    function( n, m );
  }
  else {
    cout << "the end" << endl;
  }
}

int main() {
  function( 1, 4 );
}
```

2. Is the function above tail recursive? Why/why not?

## Exercise 6

The following questions refer to the following class.

```
class Node {
  ....
  private:
    int Value;
    Node *next;
};
```

Consider the following code:

```
Node ANode;
Node *p = new Node(25, NULL); // allocate a node
Node *q;
Node **tp;
Node *NodeArray[MaxNodes];
```

1. Write a line of code that sets *tp* to point to *q*.
2. Assume that *tp* has been set to point to *q*. Write a line of code that makes *q* point to what *p* points to using *tp*. That is, don't assign to *q* directly. Change *q* by assigning to *tp*.
3. Write a line of code that sets *q* to point to the *Node ANode*.
4. Write a for loop that initializes the array *NodeArray* so each element points at the node *p* is pointing to.
5. Write a line of code that sets *tp* to point to the third element of *NodeArray*.