

Lab One

Reynaldo Alvarez

Reynaldo.Alvarez1@Marist.edu

February 2, 2018

1 PROBLEM ONE - DRAGON

1.1 LANGUAGE PROCESSORS

A compiler is a program that can read a specific *source* language and then translate that language in order for the code to be understood by a language that is different than the *source* code, which is called the *target* language.

A compiler allows code in a specific language to be translated and ran in another. For example, if we would like Java source code to be understood and ran by Windows, Mac, or Linux then we would use a compiler to make that happened.

1.2 EXERCISE 1.1.4

A compiler that translates a high-level language into another high-level language is called a *source-to-source* translator. What advantages are there to using C as a target language for a compiler?

1.2.1 ANSWER FOR 1.1.4

1.3 EXERCISE 1.6.1

For the block-structured C code of Fig. 1.13(a), indicate the values assigned to w, x, y, and z.

1.3.1 ANSWER FOR 1.6.1

2 PROBLEM TWO

3 PROBLEM THREE

4 APPENDIX

4.1 SOME JAVASCRIPT SOURCE CODE LISTINGS

```
1 var A = [5,0,5,6,6,8,45,50];
2
3 function solve(A) {
4     // Base case to stop the recursion.
5     if (A.length == 1) {
6         if (A[0] % 5 == 0) {
7             var retVal = 1;
8         } else {
9             var retVal = 0;
10        }
11        return retVal;
12    } else {
13        // Divide.
14        var divPoint = Math.floor( A.length / 2);
15        var left  = A.slice(0, divPoint);
16        var right = A.slice(divPoint, A.length);
17
18        // Conquer.
19        var left5s  = solve(left, level+1);
20        var center5s = straddle(left, right);
21        var right5s = solve(right, level+1);
22
23        // Combine.
24        return Math.max(left5s, Math.max(center5s, right5s));
25    }
26 }
27
28 function straddle(left, right) {
29     var retVal = 0;
30     if ((left[left.length-1] % 5 == 0) && (right[0] % 5 == 0)) {
31         // Count back the 5's on the left going from right to left.
32         var leftCount = 0;
33         var index = left.length-1;
34         while ( (index >= 0) && (left[index] % 5 == 0) ) {
35             index--;
36             leftCount++;
37         }
38         // Count forward the 5's on the right going from left to right.
39         var rightCount = 0;
40         while ( (rightCount < right.length) && (right[rightCount] % 5 == 0) ) {
41             rightCount++;
42         }
43         // Return the sum of the straddling 5s on the left and right.
44         retVal = leftCount + rightCount;
45     }
46     return retVal;
47 }
```