California State University, Fullerton
Dept of Computer Science

CPSC 323 Compilers & Languages (Spring 2024)

ASSIGNMENT 3

Please answer the questions appropriately, and if you believe a diagram is required, draw one.

If you want to solve the problems, do so while documenting your steps.

NOTE: Please avoid writing single-word responses or just the single solution to a problem. Please submit a PDF document.

1. Explain Three-Address Code (TAC). Convert a = (c * b) + (c * d) into three address code.

2. Convert the expression into Postfix notation:
   A + B * C / D – E

3. What is an Abstract Syntax Tree (AST)? How is an AST different from a parse tree?

4. How does constant folding contribute to improved code performance?

5. What is dead code elimination. Give an example.

6. How can reduction in strength code optimization be applied to the following example? Justify.

```
#include <stdio.h>

void doubleArrayElements(int array[], int n) {
   for (int i = 0; i < n; i++) {
      array[i] = array[i] * 2;
   }
}
```

7. Examine this code snippet:

```
#include <stdio.h>

int sum_array(int arr[], int n) {

   int sum = 0;

   for (int i = 0; i < n; i++) {

      sum += arr[i];

   }

   return sum;

}
```

   In this example, which expression is loop invariant expression? How does recognizing loop invariant expressions contribute to reducing redundant computations?

8. How does peephole optimization contribute to code size reduction? Explain with an example.

9. What is live analysis in the context of compiler optimization?

10. What is the difference between LEX and YACC. Explain with the help of block diagrams.