

Assignment #2

CS3300

1. [5] **Activation records:** Draw the runtime activation records at the entry and exit to each function. Pay special attention to the following fields of the activation records: actual parameters, return value, control link, and local variables.

```
int foo(){
    int x[] = {2, 3, 4}, len = 3;
    return sum(x, len, 0);
}
int sum(int *x, int len, int sum){
    int i, v;
    if (len <= 0) return sum;
    return sum(x+1, len-1, sum+x[0]);
}
```

2. [5] **Code generation:** Write an SDT to generate IR in three-address code (similar to the one discussed in the class) for the following grammar. Briefly explain about the attributes you use.

```
P → S
S → WhileStmt | Assignment | S; S | ε
WhileStmt → while ( Id ) { Stmt2 }
Stmt2 → S | Break
Assignment → x = E;
E : RelEx | AddEx | Id
RelEx → E < E
AddEx → E + E
Break → break
```

3. [5] **Flow-graph:** Generate the three-address code and draw the control flow graph.

```
void mm (int A[], int B[], int C[], int n){
    for (int i=0;i<n;++i)
        for (int j=0;j<n; ++j) {
            C[i][j] = 0;
            for (int k=0;k<n; k=k+4) {
                C[i][j] += A[i][k]*B[k][j];
                C[i][j] += A[i][k+1]*B[k+1][j];
                C[i][j] += A[i][k+2]*B[k+2][j];
                C[i][j] += A[i][k+3]*B[k+3][j];
            }
        }
}
```

4. [5] **Register Allocation:** Prove that the live variable analysis discussed in the class computes the liveness information for each variable, conservatively. At runtime, if the value of a variable v is live at a program point L (that is used at a later point of L), then the computed $In(L)$ set includes v .
5. [5] **Personal!** Based on the portion covered till the assignment release date (in CS3300), make an interesting question and attempt an answer for the same. Credit will be given based on the creativity, ingenuity, and coolness of the question and the answer.