BRAC UNIVERSITY
Department of Computer Science and Engineering

Examination: FINAL | Semester: Fall 2023
Duration: 1 hour 30 minutes | Full Marks: 30

CSE 420: Compiler Design

Figures in the right margin indicate marks.

Answer all the questions

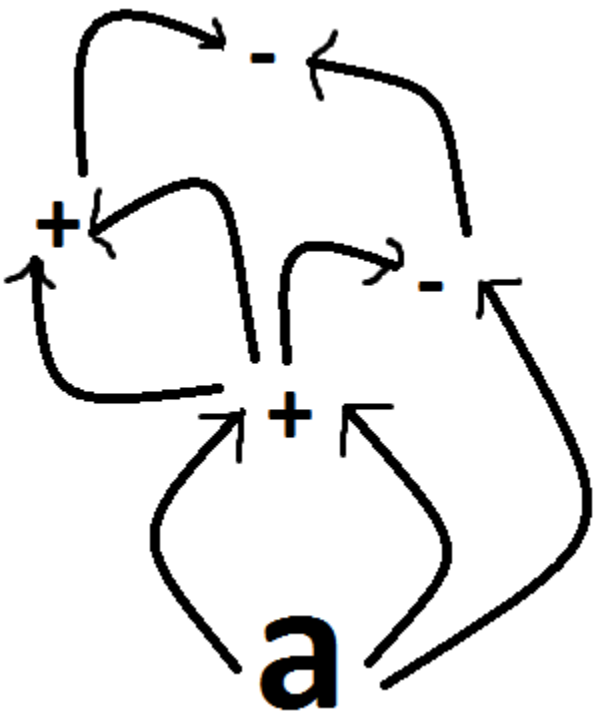
COs	Questions	Marks
CO4	**1.** Consider the following *SDT* for type declarations:	10

$$
\begin{array}{lcll}
P & \rightarrow & M\ D & \\
M & \rightarrow & \epsilon & \{\ offset = 0;\ \} \\
D & \rightarrow & T\ \mathbf{id}\ ; & \{\ top.put(\mathbf{id}.lexeme,\ T.type,\ offset); \\
 & & & \quad offset\ =\ offset + T.width;\ \} \\
 & & D_1 & \\
D & \rightarrow & \epsilon & \\
T & \rightarrow & \mathbf{record}\ '\{' & \{\ Env.push(top);\ top = \mathbf{new}\ Env(); \\
 & & & \quad Stack.push(offset);\ offset = 0;\ \} \\
 & & D\ '\}' & \{\ T.type = record(top);\ T.width = offset; \\
 & & & \quad top = Env.pop();\ offset = Stack.pop();\ \} \\
T & \rightarrow & B & \{\ t = B.type;\ w = B.width;\ \} \\
 & & C & \{\ T.type = C.type;\ T.width = C.width;\ \} \\
B & \rightarrow & \mathbf{int} & \{\ B.type = integer;\ B.width = 4;\ \} \\
B & \rightarrow & \mathbf{float} & \{\ B.type = float;\ B.width = 8;\ \} \\
C & \rightarrow & \epsilon & \{\ C.type = t;\ C.width = w;\ \} \\
C & \rightarrow & [\ \mathbf{num}\ ]\ C_1 & \{\ C.type\ =\ array(\mathbf{num}.value,\ C_1.type); \\
 & & & \quad C.width = \mathbf{num}.value \times C_1.width;\ \}
\end{array}
$$

	Fig 1: SDT for type declaration **a.** Determine the types and relative addresses for the identifiers in the following declaration by drawing the annotated parse tree and evaluating the attribute values of each non- terminal of the SDT: $record\ \{\ float\ [2][3]\ a;\ int\ b;\ \}\ p;$	
CO4	**2.** Given the following C/C++ code below, suppose the CPU is executing the line '$r = r * m$' in the code. Draw the content of the active symbol tables at that point. Show the organization of the active symbol tables as a *single list of hash-tables* in your drawing and for each variable in a symbol table, write its name and type. Remember that, in C/C++ each {} group represents a separate scope. That is, there are separate scopes for functions, for, and while loops. You do not need to separate function header and body into two separate scopes, by the way. (Hint, when you write a statement like '$t\ n\ =\ v$' in C, it means you are defining a new variable with the name n of the type t and assigning it an initial value v)	10

```
float x = 0.0;
float v = 0.0;
int[10][20] m;
   void main(int argCount, char*[] argValues) {
    x = toFloat(argV[1]);
    m = readMatrixFromFile(argV[2]);
    float t = 1.0;
    int r = 10;
          for (int i = 0; i < r; i += 1) {
          float s = 0.0;
          int c = 20;
          for (int j = 0; j < c; j += 1) {
            int v = m[i][j];
            int count = 0;
            float r = 1.0;
            while (count < v) {
                count += 1;
                float m = sqrt(x);
                r = r * m;
                }
            s = s + r;
            }
```

	println(s);
	t = t * s;
	}
	println(t);
	}
	Fig 2: A sample C/C ++ code	

CO5	**3**. Consider the following *Directed Acyclic Graph (DAG)*  Fig 3: A *Directed Acyclic Graph* **a.** Write 3 address code for the above DAG **b.** Show *quadruples* and *triples* representation of the 3 address code	10