UCT CSC305 2004 :: Compilers :: Test 2 [35 marks] :: 5 May

Error Recovery

1. Describe Burke-Fisher error repair. [4]

Abstract Syntax Trees

2. Describe the Visitor pattern and its use. [4]

Symbol Tables

3. What is a symbol table? Give one example of the type of problem it helps to solve when writing a compiler. [4]
4. In terms of non-local name resolution, what is the difference between static and dynamic scope? [2]
5. Explain how entries in a recently closed scope (assuming static scope) can be removed from an imperatively designed symbol table, implemented as a hash table. Draw a diagram to support your explanation. [5]

Activation Records

6. What is an activation record? [2]
7. With non-reentrant subprograms, why is a stack not necessary for activation records? [2]
8. Draw the stack of activation records corresponding to the following Pascal-like program when it is at “breakpointX”. [5] (Assume static chains and include all parameters).

```
program main ()

   subprogram funca ()
   {
      funcb ();
   }
   subprogram funcb ()
   {
      subprogram funcc ( int x )
      {
         x = x + 1;
      }
      funcc (6);
      // breakpointX
   }
   funca ();
}
```

Intermediate Representations

9. Assuming the IR tree language in the attached page, convert the following statements/expressions to equivalent IR trees. (Assume a and b are stack frame variables at offsets k0 and k1 respectively from the frame pointer special temporary *fp*) Provide the final trees and do not use the Nx/Cx/Ex expression types/objects. [8]

   a. a+b

   b. while (a<1) { b = b + 1; }