John Pennig

Homework 7

Review Questions:

3. **What is an alias?** Aliases are when multiple variables refer to the same address. These multiple variables that refer to the same address are aliases. An example of an alias would be a variable called count which counts the number of times something is happening in the program and iterations which could've both been initialized to the same place in memory.

5. **What is the *l*-value of a variable? What is the *r*-value?** The address of a variable is called its l-value. The r-value is the variables value.

6. **Define *binding* and *binding time*.** A binding is an association between an attribute and an entity such as a variable and its type. Binding time refers to when the binding takes place.

7. **After language design and implementation, what are the four times bindings can take place in a program?** The four times bindings can take place in a program are compile, load, link, and run time.

8. **Define *static binding* and *dynamic binding*.** Static binding refers to when binding occurs before run time and does not change during execution. Dynamic binding can occur during run time or is considered dynamic if it is able to change during runtime.

10. **What are the advantages and disadvantages of dynamic type binding?** An advantage of dynamic type binding is that you can declare a variable dynamic when the data is unknown and that you can bind variables to any type. The disadvantage of dynamic type binding is it causes programs to be less reliable because it is harder to detect errors with dynamic types. Another disadvantage of dynamic type binding is that the cost of implementing dynamic types is great. An example of dynamic type binding is in python, you do not have to declare types for functions, and they can accept any time and return any type.

11. **Define *static*, *stack-dynamic*, *explicit heap-dynamic*, and *implicit heap-dynamic variables*. What are their advantages and disadvantages?**

Static: A static variable is one where it is bound to a memory cell and remains bound to that same memory cell until the program terminates. An advantage of static variables is efficiency. A disadvantage of static variables is reduced flexibility, static variables cannot support recursive subprograms, and storage cannot be shared among variables.

Stack-Dynamic: Stack-dynamic variables are those whose storage bindings are created when their declaration statements are elaborated, but the types are statically bound. An advantage of stack-dynamic variables is the ability to have a dynamic storage so that each recursive subprogram has its own version of the local storage. A disadvantage of stack-dynamic variables is the run-time overhead of allocation and deallocation.

Explicit Heap-Dynamic: Explicit heap-dynamic variables are ones who are abstract memory cells that are allocated and deallocated by explicit run-time instructions written by the programmer. The variables are allocated and deallocated from a heap which is a collection of storage cells. An advantage of explicit heap-dynamic variables is the ability to create linked-lists and trees that need to grow and shrink during execution. The disadvantage of explicit heap-dynamic variables is handling pointers and references correctly, the cost of references, and the complexity of the required storage management implementation.

Implicit Heap-Dynamic: Implicit heap-dynamic variables are only bound to heap storage when they are assigned values. An advantage of implicit heap-dynamic variables is that they have the highest level of flexibility. A disadvantage of implicit heap-dynamic variables is the overhead of maintaining all the dynamic attributes.

12. **Define *lifetime*, *scope*, *static scope*, and *dynamic scope*.**

Lifetime: The lifetime of a variable is the time during which it is bound to a specific cell.

Scope: The scope of a variable is the range of statements in which the variable is visible.

Static Scope: Static scoping is the method of binding names to nonlocal variables. Static scoping happens prior to execution.

Dynamic Scope: Dynamic scope is based on the calling sequence of subprograms, not on their spatial relationship. Meaning that the program looks at the most recent assignment to the variable on the stack rather than the space in the program.

15. **What is the referencing environment of a statement?** A referencing environment of a statement is the collection of all variables available in the statement.

18. **What is the purpose of the `let` constructs in functional languages?** The let construct in functional languages is used to bind variables to values in a local scope.

Problem Set:

8. Global variables of x, y, and z.

In the function sub1, the local variables are a, y, and z. The visible variables are x, y, z, and a. This function has redeclarations of y, and z.

In the function sub2, the local variables are a, b, and z. This function is in the sub1 scope. This function can also see the global variable of x. This function has redeclarations of a, and z.

In the function sub3, the local variables are a, x, and w. This function can see the global variables of y, and z. This function has a redeclaration of the variable x.

9. Global variables of x, y, and z.

Sub1: The local variables are a, y, and z. This function can see the global variable of x. The function has redeclarations for y, and z.

Sub2: The local variables are a, x, and w. This function can see the global variables of y, and z. This function has a redeclaration of the variable x.

Sub3: The local variables of this function are a, b, and z. This function can see the variable of a in sub2. The function has redeclarations for z.

10. 1. The visible variables are b, c, d, and a. The a is from definition 1. The b, c, and d are from definition 2.

2. The visible variables are c, d, e, b, and a. The c, d, and e are from definition 3. The b is from definition 2. The a is from definition 1.

3. The visible variables are a, b, c, and d. The a is from definition 1. The b, c, and d are from definition 2.

4. The visible variables are a, b, and c. The a, b, and c are from definition 1.

11. a. The variables d, e, and f are local variables to fun3 and are visible. The variable c is visible and is a variable of fun2. The variable b is visible and is a variable of fun1. The variable a is visible and is a variable of the main function.

b. The variables d, e, and f are local variables to fun3 and are visible. The variables b, and c are visible and are variables of fun1. The variable a is visible and is a variable of the main function.

c. The variables b, c, and d are local variables to fun1 and are visible. The variables e, and f are visible and are variables of fun3. The function fun2 is called but all the variables are redeclared. The variable a is visible and is a variable of the main function.

d. The variables b, c, and d are local variables to fun1 and are visible. The variables e, and f are visible and are variables of fun3. The variable a is visible and is a variable of the main function.

e. The variables c, d, and e are local variables to fun2 and are visible. The variable f is visible and is a variable of fun3. The variable b is visible and a variable of fun1. The variable a is visible and is a variable of the main function.

f. The variables b, c, and d are local variables of fun1 and are visible. The variable e is visible and is a variable of fun2. The variable f is visible and is variable of fun3. The variable a is visible and is a variable of the main function.

12. a. The variables a, x, and w are local variables of sub3 and are visible. The variables b, and z are visible and are variables of sub2. The variable y is visible and are visible and is variable of sub1.

b. The variables a, x, and w are local variables of sub3 and are visible. The variables y, and z are visible and are variables of sub1.

c. The variables a, y, and z are local variables of sub1 and are visible. The variables x, and w are visible and are variables of sub3. The variable b is visible and is a variable of sub2.

d. The variables a, y, and z are local variables of sub1 and are visible. The variables x, and w are visible and are variables of sub3.

e. The variables a, b, and z are local variables of sub2 and are visible. The variables x, and w are visible and are variables of sub3. The variable of y is visible and is a variable of sub1.

f. The variables a, y, and z are local variables of sub1 and are visible. The variable of b is visible and is a variable of sub2. The variables x and w are visible and are variables of sub3.