Name: Yichen Huang

CS403/503 Programming Languages
Spring 2021
Assignment #2

1. Problem 7 of Chapter 5 on Page 229

Answer:

Static scoping: x = 5

Dynamic Scoping: x = 10

2. Problem 10 of Chapter 5 on Page 231

Answer:

Point 1:

a : 1

b : 2

c : 2

d : 2

Point 2:

a : 1

b : 2

c : 3

d : 3

e : 3

Point 3:

a : 1

b : 2

c : 2

d : 2

Point 4:

a : 1

b : 1

c : 1

3. Problem 11 (b-f) of Chapter 5 on Page 231

Answer:

	Variable	Where Declared
(b)	a	main
	b, c	fun1
	d, e, f	fun3
(c)	a	main
	e, f	fun3
	b, c, d	fun1
(d)	a	main
	e, f	fun3
	b, c, d	fun1
(e)	a	main
	b	fun1
	f	fun3
	c, d, e	fun2

(f)	a	main
	f	fun3
	e	fun2
	b, c, d	fun1

4. Review Question 37 of Chapter 6 on Page 296

Answer:

Lazy approaches (Mark-sweep):

Consists of three distinct phases. First, all cells in the heap have their indicators set to indicate they are garbage. The second part called the marking phase. Every pointer in the program is traced into heap, and all reachable cells are marked not being garbage. The third phase called sweeping phase which all cells in the heap that not marked will be returned to the list of available list.

Eager approaches (Reference counters):

Maintain a counter in every cell that store the number of pointers currently pointing at the cell. It is intrinsically incremental, so significant delays in the application execution are avoided. However, the space required, execution time required, complications for cell connected circularly.

5. Problem 4 of Chapter 6 on Page 297

Answer:

Tombstone:

Tombstones are costly in both time and space. When heap-dynamic variable deallocated, the tombstone remains but set to null. When access the heap dynamic variable through the tombstone, it requires the computer run additional machine cycle which cost time. Apparently, none of the designers of more popular language have found additional safety to be worth the cost because no widely used language use tombstones.

Lock-and-Key:

Heap-dynamic variables are represented as variable and a cell for lock value which is integer. When heap-dynamic variable allocated, lock value is created and place in the lock cell and key cell of the pointer. When the heap-dynamic variable deallocated the lock value will be cleared and the access will not be allowed, which cost less space than tombstone.