Homework 3

Keizou Wang

February 27, 2025

1. Assume that each integer variable occupies four bytes. How much total space is required for the variables in this code? Justify your answer.

```
{ int a, b, c;
     ...
     { int d, e;
          ...
          { int f;
               ...
          }
          ...
     }
     ...
     { int g, h, i;
          ...
     }
     ...
}
```

The variables will occupy 24 bytes total because up to 6 variables exist in a given scope at a time.

2. Assuming static scope, what is the referencing environment (i.e., what names are known, and what do they refer to) at the location marked by (*)?

```
procedure P (A,B: real)
    X: real
    procedure Q (B,C: real)
        Y: real
        ...
    procedure R (A,C: real)
        Z: real
        ... // (*)
    ...
```

A, B, C, X, and Z are known. B refers to P's parameter. X refers to the variable defined in P. A, C refer to R's parameters. Z refers to the variable defined in R.

3. Consider the following pseudocode:

```
1. procedure main
2.   a: integer := 1
3.   b: integer := 2

4.   procedure middle
5.     b: integer := a

6.     procedure inner
7.       print a, b

8.     a: integer := 3

9.     // body of middle
10.    inner()
11.    print a, b

12.  // body of main
13.  middle()
14.  print a, b
```

(*a*) Suppose this was code for a language with the declaration-order rules of C (but with nested subroutines). At each print statement, indicate which declarations of a and b are in the referencing environment. What does the program print?

Line 7

- a defined at line 2
- b defined at line 5

Line 11

- a defined at line 8
- b defined at line 5

Line 14

- a defined at line 2
- b defined at line 3

Program output:

```
1 1
3 1
1 2
```

(*b*) Repeat the exercise for the declaration-order rules of C#.

Line 7

- a defined at line 8

b defined at line 5

Line 11

a defined at line 8

b defined at line 5

Line 14

a defined at line 2

b defined at line 3

Program returns error because a is used before definition at line 5.

(*c*) Repeat the exercise for the declaration-order rules Modula-3.

Line 7

a defined at line 8

b defined at line 5

Line 11

a defined at line 8

b defined at line 5

Line 14

a defined at line 2

b defined at line 3

Program output:

```
3 3
3 3
1 2
```

4. Consider the following pseudocode:

```
x: integer := 1
y: integer := 2
procedure add
    x := x + y
procedure second (P: procedure)
    x: integer := 2
    P()
procedure first
    y: integer := 3
    second(add)
// main program
first()
write integer(x)
```

(a) What does this program print if the language uses static scoping?

`3`

(b) What does it print if the language uses dynamic scoping with deep binding?

`4`

(c) What does it print if the language uses dynamic scoping with shallow binding?

`1`