

CvP Practice Class 3

Chapter 5: Review Questions

3. In what way are reserved keywords better than keywords?
4. What is an alias?
6. What is the *l*-value of a variable? What is the *r*-value?
8. After language design and implementation, what are the four times bindings can take place in a program?
9. Define *static binding* and *dynamic binding*.
12. Define *static*, *stack-dynamic*, *explicit heap-dynamic*, and *implicit heap-dynamic variables*. What are the advantages and disadvantages of these?
13. Define *coercion*, *type error*, *type checking*, and *strong typing*.
14. Define *name type compatibility* and *structure type compatibility*. What are the relative merits of the two?

Chapter 5: Problem Set

8. Consider the following Ada skeletal program:

```
procedure Main is
  X: Integer;
  procedure Sub3; --This is a declaration of Sub3
                 --It allows Sub1 to call it
  procedure Sub1 is
    X: Integer;
    procedure Sub2 is
      begin --of Sub2
        ...
      end; --of Sub2
    begin
      ...
    end; --of Sub1
  procedure Sub3 is
```

```

begin --of Sub3
    ...
end --of Sub3

begin --of Main
    ...
end; --of main

```

Assume that the execution of this program is in the following unit order:

1. Main calls Sub1
 2. Sub1 calls Sub2
 3. Sub2 calls Sub3
- a. Assuming static scoping, in the following which declaration of X is the correct one for a reference to X?
- i. Sub1
 - ii. Sub2
 - iii. Sub3
- b. Repeat part a, but assume dynamic scoping.
9. Assume the following Ada program was compiled and executed using static-scoping rules. What value of X is printed in procedure Sub1? Under dynamic-scoping rules, what value of X is printed in procedure Sub1?

```

procedure Main is
X: Integer;
procedure Sub1 is
begin --of Sub1
    Put(X);
end; --of Sub1

procedure Sub2 is
X: Integer;
begin --of Sub2
    X := 10;
    Sub1
end --of Sub2

begin --of Main
    x := 5;
    Sub2
end; --of main

```

12. Consider the following C program:

```

void fun(void) {

```

```

int a,b,c; /* definition 1 */

...
while (...) {
  int b,c,d; /*def. 2 */
  ... <----- 1
  while (...) {
    int c,d,e; /*def. 3 */
    ... <----- 2
  }
  ... <----- 3
}
... <----- 4
}

```

For each of the four marked points in this function, list each visible variable, along with the number of the definition statement that defines it.

14. Consider the following program:

```

procedure Main is
  X, Y, Z : Integer;

  procedure Sub1 is
    A, Y, Z : Integer;
    begin -- of Sub1
      ...
    end; -- of Sub1

  procedure Sub2 is
    A, B, Z : Integer;
    begin -- of Sub2
      ...
    end; -- of Sub2

  procedure Sub3 is
    A, X, W : Integer;
    begin -- of Sub3
      ...
    end; -- of Sub3

begin --of Main
  ...
end; -- of Main

```

Given the following calling sequences and assuming that dynamic scoping is used, what variables are visible during execution of the last subprogram activated? Include with each visible variable the name of the unit where it is declared.

1. Main calls Sub1; Sub1 calls Sub2; Sub2 calls Sub3.
2. Main calls Sub1; Sub1 calls Sub3.

3. Main calls Sub2; Sub2 calls Sub3; Sub3 calls Sub1.
4. Main calls Sub3; Sub3 calls Sub1.
5. Main calls Sub1; Sub1 calls Sub3; Sub3 calls Sub2.
6. Main calls Sub3; Sub3 calls Sub2; Sub2 calls Sub1.