

Chapter 5 Summary

- Understand attributes of variables in the aspect of the language design.
- Understand storage of variables.
- Understand the concepts of scoping and variable life time.
- Understand dynamic typing and static typing variables as well as dynamic and static type checking.
- Understand static scoping and dynamic scoping concepts.

Exercise: Chapter 5

1. Compare and contrast the variable names of at least two languages that you know.
2. What is alias? Is the feature good or bad in the aspects of readability and writability?
3. What are the possible binding times? Give examples.
4. Why variables are important features in imperative languages?
5. Give 4 examples of variable attributes.
6. How is dynamic binding different from static binding?
7. What are the four kinds of storage binding for variables? How the storage binding differs from the variable life time? Give some example.
8. What are the possible ways of type compatibility? Explain them.
9. Given examples of the variables in C that are from the following storage:
 - (a) Static memory
 - (b) Heap dynamic memory
 - (c) Stack dynamic memory
10. Given an example of statements in the scripting language that has dynamic typing.
11. What are pros and cons of dynamic typing and static typings?
12. Consider the following pascal-like code.

What will be printed out if

- (a) static scoping is used. _____
- (b) dynamic scoping is used. _____

```
proc main
var x,y,z;
proc sub1
  var x,z
  x := 5;
  z := 7;
  sub2;
  x := y*z +x;  ----- (1)
end
proc sub2
var x,y
x := 1;
y := 1;
y := x+z+y;
print (y);
end;
begin
  x :=2; y :=3; z := 4;
  sub1;
end.
```

What are the referencing environment of point (1) if

- (i) Static scoping is used.
- (ii) Dynamic scoping is used.

13. Consider the following code and answer the questions.

```
int g=3; // 1
int f1()
{
    int x=1; // 2
    static int y=2; // 3
    y = y+2;
    x = x+1+y;
    return x+y;
}
main()
{
    int *p; // 4
    int z; // 5
    z = f1();
    z = f1() + z;
    p = (int*) malloc (sizeof(int)); // 6
    *p = f1();
    z = z + *p + g;
    printf("%d ",z); // 7
    free(p); // 8
}
```

- (a) What are the types of binding at location 1,2 and 6 ?
- (b) List all variables visible at line 8.
- (c) What is printed at line 7?

