C Programming & Lab

5. Selection Statements

Sejong University

Outline

1) **Selection Statements**
2) **if statement**
3) **if~else statement**
4) **Cascaded if statement**
5) **switch statement**

1) Selection Statements

- **Control Statement**
  - C programming language sequentially processes the source code from the first line (if not specifically stated)
  - May need to change the linear flow of the source code → Control Statements

- **Types**

Type	Examples
Selection Statements	**if statement, if~else statement, switch statement**
Loop Statements	for statement, while statement, do~while statement
Others	**break statement**, continue statement, goto statement, return statement

1) Selection Statements

- **Selection Statements**
  - Choose between two or more execution paths
  - if statement, if-else statement, switch statement

  - Examples

```
Given an integer value:
if positive (>0), print the value,
if negative (<0), print -1.
```

2) if statement

- **if statement**
  - If expression is true, execute statement1
  - Syntax

```
if( expression ) {
    statement1
}
```

  - Example) if x is larger than 0, print “positive number."

```
if( x > 0 ) {
     printf(“positive number.\n");
}
```

2) if statement

- **Write if statements for each:**

  1. If a variable x is smaller than or equal to 0, then increase the value of x by 1
  2. If a variable x is larger than -10 and smaller than 10, then assign 0 to x
  3. If a variable num is equal to 0, print “it is zero"

2) if statement

- **(Example1) Read a number from a user, determine if the number is a multiple of 2, 3, or 5.** (Hint: use 3 if statements)

Input

```
60
```

Output

```
Multiple of 2
Multiple of 3
Multiple of 5
```

- **(Example2) Read two numbers from a user, determine which number is larger or they are the same** (Use 3 if statements)

Input

```
2 3
```

Output

```
Larger number is 3
```

Input

```
4 3
```

Output

```
Larger number is 4
```

Input

```
21 21
```

Output

```
They are the same
```

2) if statement (Example3)

- **Read a character from a user, if it is a lower-case letter, then print "lower-case". If it is an upper-case letter, print "upper-case".**
  1) Use ASCII Table as shown below
  2) Do not use ASCII table, how can you do it?

Character	Decimal
A	65
B	66
Z	90
a	97
b	98
z	122

2) if statement

- **In if statement, if there is only one statement to execute (expression is true), then { } can be avoided**

```
#include <stdio.h>
int main(void){
    int x = 0;
    printf(“Enter Integer : ");
    scanf("%d", &x);
    if(x > 0)
        printf(“It is positive\n");
    return 0;
}
```

```
Enter Integer: 20
It is positive
```

```
Enter Integer: -20
```

```
if(x > 0) {
    printf(“It is positive\n");
}
```

=

```
if(x > 0)
    printf(“It is positive\n");
```

2) if statement

- **Indentation**
  - If x is equal to -1, what will be the output?

(Indentation)

```
if(x > 0)
    printf("Positive\n");
    printf("Greater than 0\n");
```

```
Greater than 0
```

    - ✓ If statement affects the first printf statement

  - Indentation is only to improve the readability.
    It does not affect the syntax.
    - ✓ Generally, indent the statements that are controlled by if statement. (Use tab key)
    - ✓ In main( ), indent all the statements

2) if statement

- **Caution! if statement**
  - Do not use semi-colon(;) after if statement
    - ✓ If use semi-colon(;), it is treated as an independent statement
  - If x is equal to -1, what will be the output?

```
if(x > 0) ;
    printf("It is positive\n");
```

||

```
if(x > 0)
    ;   // Empty statement
printf("It is positive\n");
```

```
It is positive
```

2) if statement

- **Common Mistakes: if statement**
  - Use = (Assignment), not == **(Relation Operator)**

```
int x = -1;
if( x = 10 )
    printf("x is %d\n",x);
```

✓ No compilation error → No syntax error

✓ Result?

```
X is 10
```

✓ Why? assign 10 to x, the value of x is assessed
(In C, if is not 0, then considered as true)

✓ Change to x = 0, what happens?

2) if statement

- **Exercise**
  - Read an integer number, print the absolute value of the integer
    - ✓ Think how to code
      - ① User enters an integer
      - ② Test the integer
      - ③ What you need to do to print the absolute value?

Example 1

```
Enter Integer: 20
Absolute Value = 20
```

Example 2

```
Enter Integer : -20
Absolute Value = 20
```

2) if statement

- **Answer**
  - Can be done in a different way

```
#include <stdio.h>
int main(void){
    int x=0;
    printf(“Enter Ingeger: ");
    scanf("%d", &x);
    if(x>=0) printf(“Absolute Value=%d\n",
x);
    if(x<0) printf(" Absolute Value =%d\n",
-x);
    return 0;
}
```

3) if-else statement

- **if-else statement**
  - If expression is true, execute statement1. If false, execute statement2.
  - Syntax

```
if(expression) {
    statement1
}
else {
    statement2
}
```

  - The previous exercise can be done as follows:

```
if(x >= 0)
    printf("Absolute Value = %d\n", x);
else
    printf(" Absolute Value = %d\n", -x);
```

3) if-else statement (example4)

- **Read two numbers (numerator, denominator) from a user, if denominator is 0, print “denominator is 0”. Otherwise, print quotient and remainder.**

Input	Output
`8 0`	`Denominator is 0`
`8 3`	`Quotient=2` `Remainder=2`

4) Cascaded if statement

- **if statement contains another if statement (including if-else statement)**

```
if(x >= 0)
    if(x%2 == 0)
        printf("Positive even number\n");
    else
        printf("Positive odd number\n");
```

✓ x is 4 → Positive even number

✓ x is 3 → Positive odd number

✓ x is -1 →

4) Cascaded if statement

- **Else statement is paired with the closest if statement**

```
if(x >= 0)
    if(x%2 == 0)
        printf(“Positive even number\n");
    else
        printf(“Positive odd number\n");
else  printf(“Negative number\n");
```

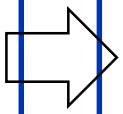
- **Else statement: pair with if statement that is not the closest one?**
  - Use { }, change the range of if statement

```
if(x >= 0) {
    if(x%2 == 0)
        printf(“Positive even number\n");
}
else printf(“Negative\n");
```

4) cascaded if statement

- **Following code is to classify an integer into 4 types. What would be the appropriate indentation for this code?**

```
if(x >= 0)
if(x%2 == 0)
printf(“Positive even number\n");
else
printf(“Positive odd number\n");
else
if(x%2 == 0)
printf(“Negative even number\n");
else
printf(“Negative odd number\n");
```



```
if(x >= 0)
    if(x%2 == 0)
        printf("Positive even number\n");
    else
        printf("Positive odd number\n");
else
    if(x%2 == 0)
        printf(“Negative even number\n");
    else
        printf(“Negative odd number\n");
```

4) Cascaded if statement(Example5)

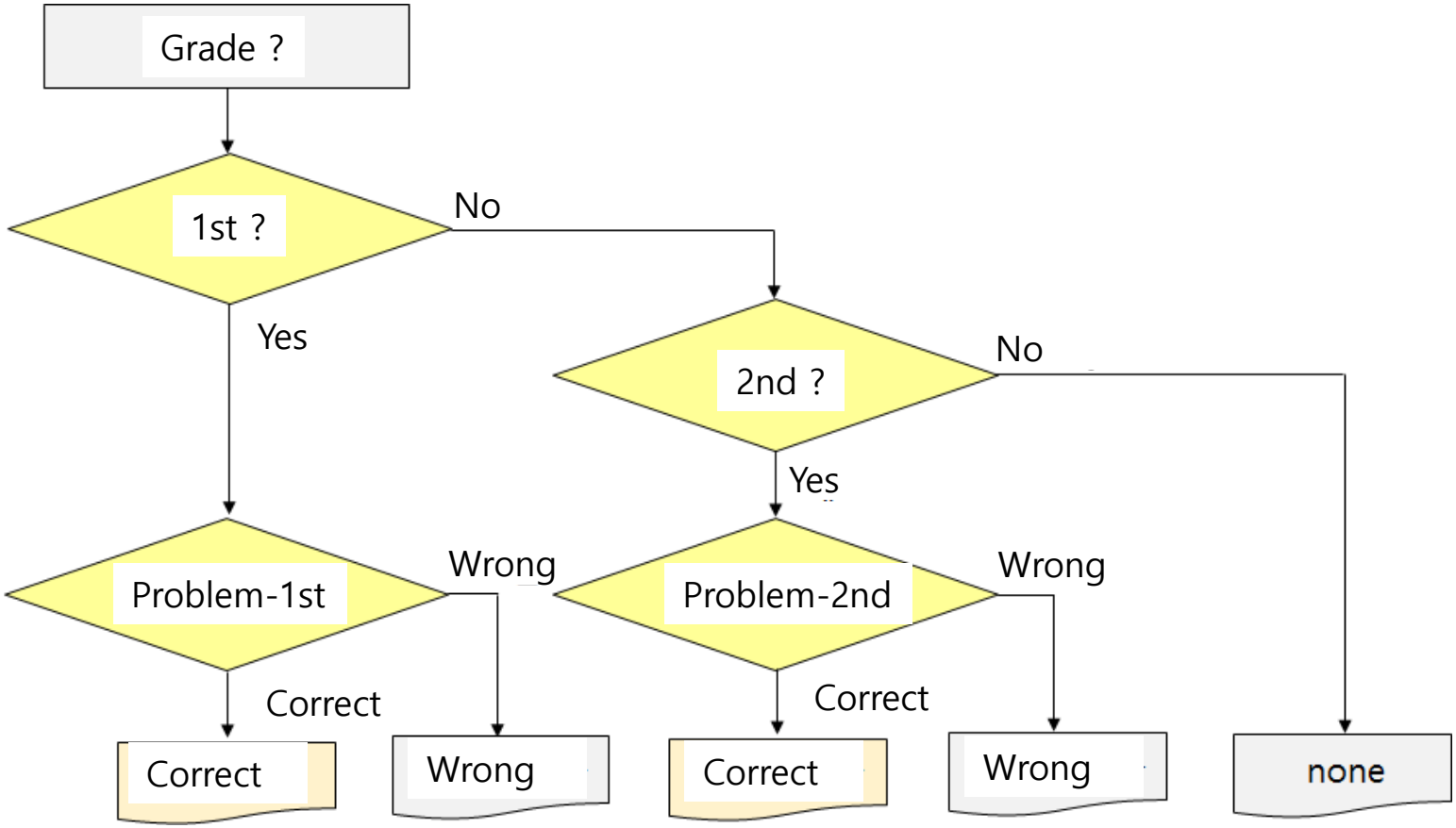
- **Write a quiz program for elementary school 1st and 2nd grades**
  - Refer the next slide

```
What grade(1, 2)? 1
Columbus discovered America. Correct 1, Wrong 0: 1
Wrong
```

```
What grade(1, 2)? 2
35 * 76 = 123
Wrong
```

4) Cascaded if statement(Example5)

- **Quiz program for elementary school 1st and 2nd grade**



4) Cascaded if statement

- **Cascaded if statement: multiple selections**
  - Write a program that assigns a grade according to the following rules

```
① Enter your score (score range 0~100)
② If greater than or equal to 90, get A
③ If greater than or equal to 80 and smaller than
   90, get B
④ If greater than or equal to 70 and smaller than
   80, get C
⑤ If greater than or equal to 60 and smaller than
   70, get D
⑥ If smaller than 60, get F
```

4) Cascaded if statement

- **Cascaded if statement: multiple selections**
  - Note: Indentation

```
#include <stdio.h>
int main(void){
    int score=0;
    printf(“Enter your score: ");
    scanf("%d", &score);
    if(score>=90) printf("A\n");
    else if(score>=80) printf("B\n");
         else if(score>=70) printf("C\n");
              else if(score>=60) printf("D\n");
                   else printf("F\n");
    return 0;
}
```

```
Enter your score: 66
D
```

4) Cascaded if statement

- **Cascaded if statement: multiple selections**
  - In this case, you may not indent the statement (clearer)

```
#include <stdio.h>
int main(void){
    int score=0;
    printf(“Enter your score: ");
    scanf("%d", &score);
    if(score>=90) printf("A\n");
    else if(score>=80) printf("B\n");
    else if(score>=70) printf("C\n");
    else if(score>=60) printf("D\n");
    else printf("F\n");
    return 0;
}
```

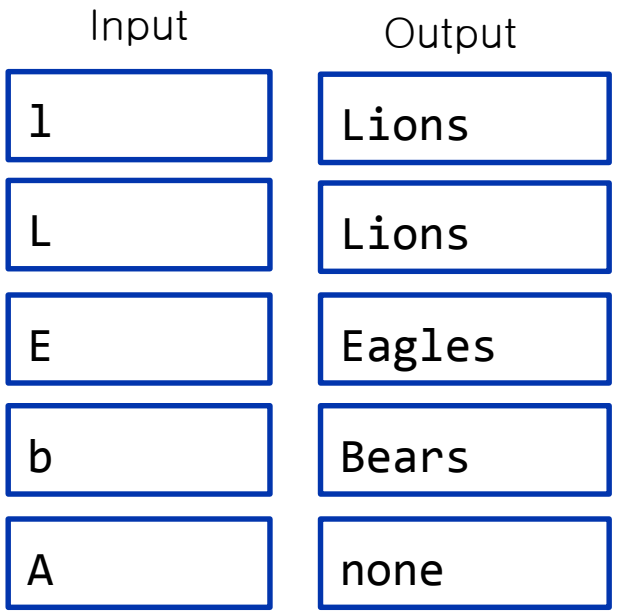
4) Cascaded if statement

- **Cascaded statement**
  - Do not indent when use multiple if ~ else statements

```
if(A) {
    statement1: A is true
}
else if(B) {
    statement2: A is false, B is true
}
else if(C) {
    statement3: A, B are false, C is true
}
else {
    statement4: A, B, C are false
}
```

4) Cascaded if statement(Example6)

- **If enter the first letter of a baseball team, print the full name of the baseball team. The letter could be either upper-case or lower-case.**



Input	Output
L or l	Lions
E or e	Eagles
B or b	Bears
others	none

4) Cascaded if statement

- **Exercise**
  - Use cascaded if statement, write a program for the following rules

```
①  Read a character from a user
②  If upper-case, print “Upper-case”.
③  If lower-case, print “Lower-case”.
④  If number, print “Number”.
⑤  Otherwise, “others”.
```

4) Cascaded if statement

- **Classification: Upper-case, Lower-case, Number, Others**

```
#include <stdio.h>
    int main(void){
    char ch;
    printf(“Enter character: ");
    scanf("%c", &ch);
    if(ch>='A' && ch<='Z') printf(“Upper-case \n");
    else if(ch>='a' && ch<='z') printf(“Lower-case \n");
    else if(ch>='0' && ch<='9') printf(“Number \n");
    else printf(“Others \n");
    return 0;
}
```

```
Enter character: *
others
```

4) Cascaded if statement

- **Note) Indent cascaded if statement properly**
  - Example) Classify an integer into 4 types (slide no. 19)
    - ✓ Which one is more intuitive?

```
if(x >= 0)
    if(x%2 == 0)
        printf(“Positive even number\n");
    else
        printf(“Positive odd number\n");
else if(x%2 == 0)
    printf(“Negative even number\n");
else
    printf(“Negative odd number\n");
```

```
if(x >= 0)
    if(x%2 == 0)
        printf("Positive even number\n");
    else
        printf("Positive odd number\n");
else
    if(x%2 == 0)
        printf("Negative even number\n");
    else
        printf("Negative odd number\n");
```

5) switch statement

- **switch statement**
  - Similar to if-else statement
  - Simpler than cascaded if statement
  - Limited usage compared to if-else statement

  - Syntax

```
switch( expression ) {
   case Constant1:
       Statement1
       break;
   case Constant2:
       Statement2
       break;
   default:
       Statement3
       break;
}
```

5) switch statement

- **Convert if~else statement to switch statement**

```
if(x==1)
    printf("January\n");
else if(x==2)
    printf("February\n");
else if(x==3)
    printf("March\n");
else
    printf("None\n");
```

```
switch(x){
    case 1:
        printf("January\n");
        break;
    case 2:
        printf("February\n");
        break;
    case 3:
        printf("March\n");
        break;
    default:
        printf("None\n");
        break;
}
```

5) switch statement

- **switch statement: how it works?**
  - Evaluate expression x in regard to the constant in the case labels one by one
    - ✓ Ignore statements if dose
    - ✓ not match
  - default: What it means?
    - ✓ Other cases
  - No matching case labels, no default: ?
    - ✓ Do not execute anything, Move to the end of switch statement

x is 3
Flow of control

```
switch(x){
    case 1:
        printf("January\n");
        break;
    case 2:
        printf("February\n");
        break;
    case 3:
        printf("March\n");
        break;
    default:
        printf("None\n");
        break;
}
```

Execute from here

5) switch statement

- **break statement**
  - Stop execution
  - Move to the end of switch statement

  - No break statement?

```
Enter a number:2
February
March
None
```

```
int main(void){
    int x=0;
    printf(“Enter a number:");
    scanf("%d", &x);
    switch(x){
       case 1: printf("January\n");
       case 2: printf("February\n");
       case 3: printf("March\n");
       default: printf("None\n");
    }
    return 0;
}
```

5) switch statement

- **case labels should be ordered as 1, 2, 3? NO!!**
  - Order does not matter! Character can be used!

```
#include <stdio.h>
int main(void){
    char ch;
    printf(“Enter grade: ");
    scanf("%c", &ch);
    switch(ch){
        case 'C': printf(“Not Bad.\n"); break;
        case 'B': printf(“Good.\n"); break;
        case 'A': printf(“Wonderful.\n"); break;
        default: printf(“Study Hard.\n"); break;
    }
    return 0;
}
```

```
Enter grade: A
Wonderful.
```

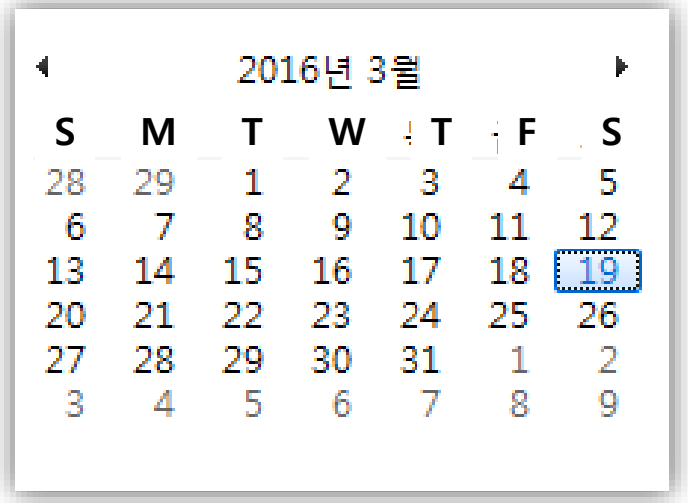
5) switch statement (Example7)

- **Use switch statement, print a grade.**
  - Hint: score/10=91/10=9, Integer/Integer=Integer

```
#include <stdio.h>
int main(void){
    int score=0;
    printf(“Enter score: ");
    scanf("%d", &score);
    if(score>=90) printf("A\n");
    else if(score>=80) printf("B\n");
    else if(score>=70) printf("C\n");
    else if(score>=60) printf("D\n");
    else printf("F\n");
    return 0;
}
```

5) switch statement (Example8)

- **Read a date and print the day of the date by using switch statement.**
  - Hint: Use modulus (%), 8 % 3 = 2




```
Enter date: 28

Monday
```

5) switch statement (Example9)

- **Obtain information from a resident registration number according to the following table**

First number	
1 or 3 or 9	Male
0 or 2 or 4	Female
5, 6, 7, 8	Foreigner

```
Enter first number in your resident registration number: 1
Male
```

```
Enter first number in your resident registration number: 5
Foreigner
```

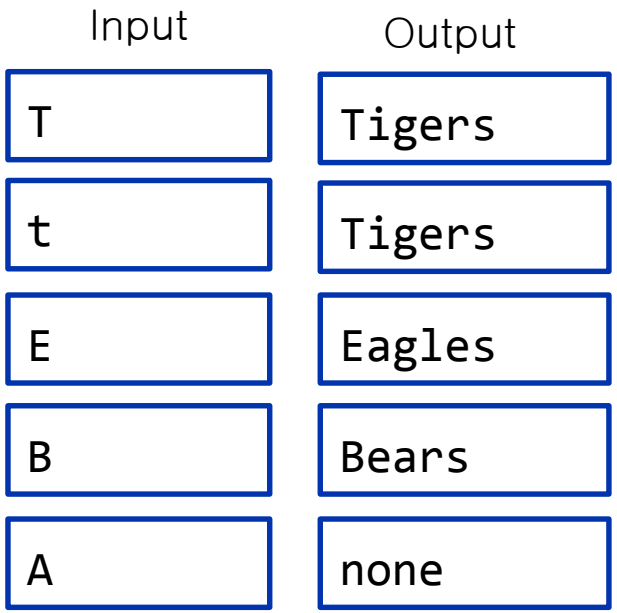
5) switch statement

- **Caution: switch statement**
  - In case label, use only integer type constants (including characters)
    - ✓ case 1 : // OK
    - ✓ case 'A' : // OK (Character is an integer in C)

    - ✓ case 2.0 : (X) // No real number.
    - ✓ case 3+4 : (X) // No expression.
    - ✓ case x<3 : (X) // No expression.
    - ✓ case A : (X) // No variable.
      // Caution!! A is variable, 'A' is character
    - ✓ case "A" : (X) // No character string.
      // Caution!! 'A' is character, "A" is character string

5) switch statement (Example10)

- **If enter the first letter of a baseball team, print the full name of the baseball team. The letter could be either upper-case or lower-case. Use switch statement**



Input	Output
T or t	Tigers
E or e	Eagles
B or b	Bears
others	none

5) switch statement (Example11)

- **There are 3 MATH problems. Read a number between 0 and 9. Select one among the 3 problems.**
  - Number and problem pairs are:

Input	Problem
0, 3, 4, 7	13 * 13 =
1, 2, 9	17 * 17 =
5, 6, 8	19 * 19 =
Others	none

Input

```
4

169
```

Output

```
13 * 13 =

Correct
```