CSE 142
Computer Programming I

Switch Statement

Overview
Concepts this lecture
The switch statement
Choosing between if and switch
Reading
Textbook sec. 4.8

Review: Conditional Control Flow
The if statement chooses one of two statements to execute before continuing
An if statement could also be used to decide whether or not to skip a statement before continuing

Multi-way Control Flow
The choice may be “multi-way” rather than simply between two alternatives
In C, if statements can be used, and sometimes a statement called the switch can be used

Multi-way Choice with if
/* How many days in a month? */
if ( month == 1 ) { /* Jan */
  days = 31 ;
} else if ( month == 2 ) { /* Feb */
  days = 28 ;
} else if ( month == 3 ) { /* Mar */
  days = 31 ;
} else if ( month == 4 ) { /* Apr */
  days = 30 ;
... /* need 12 of these */

Better...
if ( month == 9 || month == 4 || /* Sep, Apr */
    month == 6 || month == 11 ) { /* Jun, Nov */
  days = 30 ;
} else if ( month == 2 ) { /* Feb */
  days = 28 ;
} else {
  days = 31 ; /* All the rest */
}
Alternative: switch

A switch is a form of conditional statement.

It is specifically designed to be useful in multi-way choice situations.

Instead of a condition, there is a value which is tested, and a series of cases of which only one may be chosen.

Using switch

/* How many days in a month? */
switch ( month ) {
    case 2: /* February */
        days = 28;
        break;
    case 9: /* September */
    case 4: /* April */
    case 6: /* June */
    case 11: /* November */
        days = 30;
        break;
    default: /* All the rest have 31 ... */
        days = 31;
    }
printf ("There are %d days. 
", days);  

switch Statement

The syntax of switch differs from other C statements

    switch (int expression) {
        ...
        /* a series of cases */
        ...
    }

The value of the expression determines which of the cases is executed.

Cases

A case is a section of code within the switch statement. A case is executed only if the switch expression has a specified value

case value:
    /* a sequence of statements*/

The sequence is typically ended with special statement

break;

break causes the entire switch statement to end

The switch Expression

The switch expression is not a conditional expression as it is in an if statement

Only an integer expression is allowed

Most often, the expression is a single integer variable

The value of the variable determines which case is chosen

switch: Flow of Control

month = 6;
switch ( month ) {
    case 2: /* February */
        days = 28;
        break;
    case 9: /* September */
    case 4: /* April */
    case 6: /* June */
    case 11: /* November */
        days = 30;
        break;
    default: /* All the rest have 31 ... */
        days = 31;
    }
printf ("There are %d days. 
", days);
The One Big Pitfall of `switch`

```c
month = 6;
switch (month) {
    case 2: /* February */
        days = 28; /* break missing */
    case 9: /* September */
    case 4: /* April */
    case 6: /* June */
    case 11: /* November */
        days = 30; /* break missing */
    default: /* All the rest have 31 ... */
        days = 31;
}
printf ("There are %d days. 
", days);
```

### switch on char is also legal

```c
char marital_status;
...
switch (marital_status) {
    case 'm':
    case 'M':
        printf ("Married 
" );
        break;
    case 's':
    case 'S':
        printf ("Single 
" );
        break;
    default:
        printf ("Sorry, I don't recognize that code. 
" );
}
```

Summing Up

Switch is a form of conditional statement

Switch is suitable for multi-way conditions that depend upon an integer (or char) value

Pay attention to the syntax of switch

The switch and if statements are not fully interchangeable

---

Bonus Footnote

```c
char marital_status;
...
switch (marital_status) {
    case 'm':
    case 'M':
        ...
        ...
    default:
        Why should a character be allowed here, when the expression is supposed to be an integer?
        Answer: The actual machine representation of a character is a small integer.
        Most of the time, however, you should treat ints, and chars as fully different types!
```