Switch Statement

\[
\text{switch (expression) \{ }
\text{case \ value1: \ statement(s); break; }
\text{case \ value2: \ statement(s); break; }
\text{default: \ statement(s); }
\text{\}}
\]

<table>
<thead>
<tr>
<th>expression</th>
<th>evaluates to int or char 1, '1', -20, 'a', 'Z', '*' a + b</th>
</tr>
</thead>
<tbody>
<tr>
<td>case</td>
<td>Label which is a reserved word.</td>
</tr>
<tr>
<td>value1</td>
<td>a possible result of the expression above</td>
</tr>
<tr>
<td>break</td>
<td>A reserved word which terminates the switch statement</td>
</tr>
<tr>
<td>default</td>
<td>A reserved word which is &quot;catch all&quot;. Default always matches expression above</td>
</tr>
</tbody>
</table>

Example

switch (LtrGrd) {
    case 'A':
        printf("Your grade is between 90 and 100");
        break;
    case 'a':
        printf("Your grade is between 90 and 100");
        break;
    case 'B':
        printf("Your grade is between 80 and 89");
        break;
    case 'b':
        printf("Your grade is between 80 and 89");
        break;
    case 'C':
        printf("Your grade is between 70 and 79");
        break;
    case 'c':
        printf("Your grade is between 70 and 79");
        break;
    case 'D':
        printf("Your grade is between 60 and 69");
        break;
    case 'd':
        printf("Your grade is between 60 and 69");
        break;
    case 'F':
        printf("Your grade is less than 60");
        break;
    case 'f':
        printf("Your grade is less than 60");
        break;
    default:
        printf("The value of your grade is not valid");
}

Another Example

switch (CoinValue) {
    case 25:
    case 50:
    case 100:
        TotalMoney = TotalMoney + CoinValue;
        break;
    default:
        printf("Only 25, 50, and 100 cent coins are accepted");
}

What if the "break" is skipped?

Programming Project

1. Problem:
   - Write a program that asks the user to provide a number, then tells the user if this number is evenly divisible by 2, 3, and 5
2. Analysis
   - The remainder operation (%) can be used to determine if a number is evenly divisible by another number
   - If the number is 66:
     - 66 % 2 = 0 which means it is evenly divisible by 2
     - 66 % 3 = 0 which means it is evenly divisible by 3
     - etc.
Programming Project

3. Design:
   a) Prompt the user for input – Number
   b) Calculate Number % 2 – If no remainder DivBy2 is True
   c) Calculate Number % 3 – If no remainder DivBy3 is True
   d) Calculate Number % 5 – If no remainder DivBy5 is True
   e) Print the results

   • Since b, c and d are repetitive, write a function that takes number and divider, returns 1 if divisible and 0 if not

4. Coding

/*
 * Find if a given number is divisible by 2, 3, and 5
 */

#include <stdio.h>

int DivBy (int x, int a);  /* if x is divisible by a returns 1 if not returns 0 */

int DivBy2,  /* Divisible by 2 True:1 – False:0 */
DivBy3,  /* Divisible by 3 True:1 – False:0 */
DivBy5;  /* Divisible by 5 True:1 – False:0 */
/* Get the Number to be checked */

printf("This program finds if a number is divisible by 2, 3, and 5\n");
printf("Please enter a number: ");
scanf("%d", &Number);

/* Check if divisible by 2, 3, and 5 */

DivBy2 = DivBy(Number, 2);
DivBy3 = DivBy(Number, 3);
DivBy5 = DivBy(Number, 5);

/* Print the outcome */

switch (DivBy2 + DivBy3 + DivBy5) {
    case 0:
        printf("%d is not divisible by 2 3 5\n", Number);
        break;
    default:
        printf("%d is divisible by", Number);
        if (DivBy2 == 1) printf(" 2");
        if (DivBy3 == 1) printf(" 3");
        if (DivBy5 == 1) printf(" 5");
}

return(0);
/*
 * This function checks if first argument is divisible by second
 * Returns 1 if true and 0 if false
 */
int DivBy(int x, int a)
{
    if((x%a) == 0) return(1);
    else return(0);
}

5. Test:
   a) Try 13 – Expected result: None
   b) Try 66 – Expected result: 2 3
   c) Try 25 – Expected result: 5
   d) Try 60 – Expected result: 2 3 5

6. Maintenance:
   Try Adding divisibility by 4
Refactoring

- **Refactoring** is the process of changing a program's code without changing its functionality, in order to improve *quality attributes* of the software such as:
  - readability,
  - maintainability,
  - performance,
  - extensibility,
  - portability,
  - etc.

Dr. Sadık Eşmelioğlu  
CENG 111 – Fall 2009-10  

---

Programming Project Refactored

```c
/*
 * Find if a given number is divisible by 2, 3, and 5
 */
#include <stdio.h>
int DivBy (int x, int a); /* if x is divisible by a returns 1 if not returns 0 */
int main (void)
{
    int Number; /* Input number */
    /* Get the Number to be checked */
    printf("This program finds if a number is divisible by 2, 3, and 5\n");
    printf("Please enter a number: ");
    scanf("%d", &Number);
    /* Check if divisible by 2, 3, and 5 */
    if (DivBy(Number, 2) && DivBy(Number, 3) && DivBy(Number, 5))
        printf("Number is divisible by 2, 3, and 5\n");
    else
        printf("Number is not divisible by 2, 3, and 5\n");
}
```

Dr. Sadık Eşmelioğlu  
CENG 111 – Fall 2009-10  

---
Programming Project Refactored

/* Print the outcome */
printf("%d is divisible by", Number);
if (DivBy(Number, 2) == 1) printf(" 2");
if (DivBy(Number, 3) == 1) printf(" 3");
if (DivBy(Number, 5) == 1) printf(" 5");
return(0);
}

/* Checks if first argument is divisible by second, returns 1 if true and 0 if false */
int DivBy(int x, int a)
{
  if((x%a) == 0) return(1);
  else return(0);
}

Programming Project 2

/*
 * This Program presents a menu of operations to the user
 */
#include <stdio.h>

int main (void)
{
  int ans; /* User selection */

  /* Print the menu */
  printf("1. Operation 1\n");
  printf("2. Operation 2\n");
  printf("3. Operation 3\n");
  printf("4. Operation 4\n");
  printf("5. Operation 5\n");
  printf("Please Select an operation [1-5]: ")

Programming Project 2

/* Get selection */
scanf("%d", &ans);

/* Perform the selected operation */
switch (ans) {
    case 1:
        printf("Operation 1 is selected\n"); break;
    case 2:
        printf("Operation 2 is selected\n"); break;
    case 3:
        printf("Operation 3 is selected\n"); break;
    case 4:
        printf("Operation 4 is selected\n"); break;
    case 5:
        printf("Operation 5 is selected\n"); break;
    default:
        printf("Selection is not in the menu\n");
} /* Close switch */
return(0);
} /* Close main */

What if the "break" is skipped?

Exercise

- An alarm system sends the status to a file in a computer - C:\temp\AlarmStat.txt.
- The status is a number between 0 and 255, when converted to binary gives the status of all the alarm points as below:
  
a7 a6 a5 a4 a3 a2 a1 a0
  
a7 is SS - System Status
a6 is FD - Front Door
a5 is RD - Rear Door
a4 is KD - Kitchen Door
a3 is BD - Balcony Door
a2 is FW - Front Window
a1 is BW - Back Window
a0 is SW - Side Window

- In each of these bit locations
  - 1 means open
  - 0 means closed

- Write a program that will read the status from the status file and print to the screen if an alarm condition exits, and if it does, prints which points are alarmed.