3.10 Example: Toll calculation

Students:
Section 3.11 is a part of 2 assignments: CSC108 CH03.11-3.20 C3B
Includes: CA
Due: 02/25/2025, 11:59 PM EST
This assignment's due date has passed. Activity will still be recorded, but will not count towards this assignment (unless the due date is changed). See this article for more info.

# 3.11 Switch statements

## Switch statement

A **switch** statement can more clearly represent multi-branch behavior involving a variable being compared to constant values. The program executes the first **case** whose constant expression matches the value of the switch expression, executes that case's statements, and then jumps to the end. If no case matches, then the **default case** statements are executed.

PARTICIPATION ACTIVITY | 3.11.1: Switch statement.

Run | Speed

```
#define ch
int main() {
  printf("Enter ");
  switch (ch) {
    case 1:
      printf("one");
      break;
    case 2:
      printf("two");
      break;
    default:
      printf("unknown");
      break;
  }
}
```

Output

a = 1: one

a = 5: unknown

Captions

1. A switch statement can more clearly represent multi-branch behavior involving a variable being compared to constant values.
2. The program executes the first case whose constant expression matches the value of the switch expression, executes that case's statements, and then jumps to the end.
3. If no case matches, then the default case statements are executed.

Feedback?

PARTICIPATION ACTIVITY | 3.11.2: Switch statement.

numItems and userVal are int types. What is the final value of numItems for each userVal?

```
switch (userVal) {
   case 1:
      numItems = 5;
      break;

   case 2:
      numItems = 12;
      break;

   case 3:
      numItems = 15;
      break;

   default:
      numItems = 35;
      break;
}
```

1) userVal = 3.

Check | Show answer

2) userVal = 0.

Check | Show answer

3) userVal = 2.

Check | Show answer

Feedback?

### Multi-branch if-else statement

*A switch statement can be written using a multi-branch if-else statement, but the switch statement may make the programmer's intent clearer.*

```
if (dogYears == 1) {          // Like case 1
   // Print 21 years
}
else if (dogYears == 2) {     // Like case 2
   // Print 35 years
}
else if (dogYears == 3) {     // Like case 3
   // Print 37 years
}
else {                        // Like default case
   // Print unknown
}
```

## Switch statement general form

The switch statement's expression should be an integer or char. The expression should not be a string or a floating-point type. Each case must have a constant expression like 2 or 'c', a case expression cannot be a variable.

The order of cases doesn't matter assuming break statements exist at the end of each case. The earlier program could have been written with case 3 first, then case 2, then case 5, then case 1, for example (though that would be bad style).

*Good practice is to always have a default case for a switch statement. A programmer may be sure all cases are covered only to be surprised that some case was missing.*

Construct 3.11.1: Switch statement general form

```
switch (expression):
  case constantExpr1:
    // statement(s)
    break;

  case constantExpr2:
    // statement(s)
    break;


  default: // If no other case matches
    // statement(s)
    break;
```

Feedback?

Figure 3.11.1: Switch example: Estimates a dog's age in human years.

```
#include <iostream>
using namespace std;

/* Estimates dog's age in equivalent human years.
   Source: www.dogbreeds.com
*/

int main() {
   int dogHumanYears;

   cout << "Enter dog's age (in years): ";
   cin >> dogHumanYears;

   switch (dogHumanYears) {
      case 1:
         cout << "That's 14 human years." << endl;
         break;

      case 2:
         cout << "That's 23 human years." << endl;
         break;

      case 3:
         cout << "That's 24 human years." << endl;
         break;

      case 4:
         cout << "That's 28 human years." << endl;
         break;

      case 5:
         cout << "That's 32 human years." << endl;
         break;

      case 6:
         cout << "That's 37 human years." << endl;
         break;

      default:
         cout << "Human years unknown." << endl;
         break;

   return 0;
```

```
Enter dog's age (in years): 4
That's 32 human years.


Enter dog's age (in years): 17
Human years unknown.
```

Feedback?

zyDE 3.11.1: Switch statement: Numbers to words

Extend the program for dogYears to support age of 6 to 10 years. Conversions are 6:42, 7:47, 8:52, 9:57, 10:62.

Load default template... | 7 | Run

```
#include <iostream>
using namespace std;

/* Estimates dog's age in equivalent human years.
   Source: www.dogbreeds.com
*/

int main() {
   int dogHumanYears;

   cout << "Enter dog's age (in years): " << endl;
   cin >> dogHumanYears;

   switch (dogHumanYears):
      case 1:
         cout << "That's 14 human years." << endl;
         break;
```

Feedback?

## Omitting the break statement

Omitting the **break** statement for a case will cause the statements within the next case to be executed. Such "falling through" to the next case can be useful when multiple cases, such as cases 6, 1, and 2, should execute the same statements.

The following extends the previous program for dog ages less than 1 year old. If the dog's age is 0, the program asks for the dog's age in months. Within the `switch (dogHumanYears)` statement, "falling through" is used to execute the same display statement for several values of *dogAgeMonths*. For example, if *dogAgeMonths* is 0, 1 or 2, the same statement executes.

*A common error occurs when the programmer forgets to include a break statement at the end of a case's statements.*

Figure 3.11.2: Switch example: Dog years with months.

```
#include <iostream>
using namespace std;

int main() {
   int dogHumanYears;
   int dogAgeMonths;

   cout << "Enter dog's age (in years): ";
   cin >> dogHumanYears;

   if (dogHumanYears == 0) {
      cout << "Enter dog's age (in months): ";
      cin >> dogAgeMonths;

      switch (dogAgeMonths) {
         case 0:
         case 1:
         case 2:
            cout << "That's 0-14 human months." << endl;
            break;

         case 3:
         case 4:
         case 5:
         case 6:
            cout << "That's 1-7 human years." << endl;
            break;

         case 7:
         case 8:
         case 9:
            cout << "That's 5-9 human years." << endl;
            break;

         case 10:
         case 11:
         case 12:
            cout << "That's 8-13 human years." << endl;
            break;

         default:
            cout << "Invalid input." << endl;
            break;
      }
   }
   else {
      cout << "FIXME: Do earlier dog year cases." << endl;
      switch (dogHumanYears) {
      }
   }

   return 0;
```

```
Enter dog's age (in years): 0
Enter dog's age (in months): 7
That's 1-7 human years.


Enter dog's age (in years): 4
FIXME: Do earlier dog year cases.
```

Feedback?

PARTICIPATION ACTIVITY | 3.11.3: Switch statement.

userChar is a char and encodedVal is an int. What will encodedVal be for each userChar value?

```
switch (userChar) {
   case 'X':
      encodedVal = 1;
      break;

   case 'B':
      encodedVal = 2;
      break;

   case 'C':

   case 'D':
      encodedVal = 4;
      break;

   case 'E':
      encodedVal = 5;

   case 'F':
      encodedVal = 6;
      break;

   default:
      encodedVal = -1;
      break;
}
```

1) userChar = X

Check | Show answer

2) userChar = 'B'

Check | Show answer

3) userChar = C

Check | Show answer

4) userChar = 'E'

Check | Show answer

5) userChar = 'G'

Check | Show answer

Feedback?

CHALLENGE ACTIVITY | 3.11.1: Rock paper scissors. | Full screen

CHALLENGE ACTIVITY

Organize the lines of code to create a switch statement that checks nextChoice:

If the input is 1, then the output is `Rock`

If the input is 2, then the output is `Paper`

If the input is 3, then the output is `Scissors`

For any other input, the output is `Unknown`

How to use this tool

**Unused**

```
switch (nextChoice) {
cout << "Unknown" << endl;
case 3:
case 2:
cout << "Paper" << endl;
break;
}
cout << "Scissors" << endl;
break;
cout << "Rock" << endl;
break;
case 1:
default:
}
```

**main.cpp** | Load default template...

```
#include <iostream>
using namespace std;

int main() {
   int nextChoice;

   cin >> nextChoice;


   return 0;
}
```

Check

Feedback?

CHALLENGE ACTIVITY | 3.11.2: Switch statement to convert letters to Greek letters. | Full screen

CHALLENGE ACTIVITY

Organize the lines of code to create a switch statement that checks origLetter:

Ex: If the input is a or A, then the output is `Alpha`

Ex: If the input is e or E, then the output is `Beta`

For any other input, the output is `Unknown`

How to use this tool

**Unused**

```
cout << "Beta" << endl;
break;
case 'B':
cout << "Alpha" << endl;
break;
switch (origLetter) {
case 'A':
case 'a':
case 'E':
}
default:
cout << "Unknown" << endl;
break;
```

**main.cpp** | Load default template...

```
#include <iostream>
using namespace std;

int main() {
   char origLetter;

   cin >> origLetter;


   return 0;
}
```

Check

Feedback?

How was this section? | Provide section feedback

Activity summary for assignment: CSC108 CH03.11-3.20 C3B
Due: 02/25/2025, 11:59 PM EST
This assignment's due date has passed. Activity will still be recorded, but will not count towards this assignment (unless the due date is changed). See this article for more info.
Completion details

24 / 24 points
24 / 24 points submitted to Blackboard, earn