1806ICT
Programming Fundamentals

Selection II

1

1

Topics

- The Conditional Operator statement
- The `switch` statement
- Common mistakes

2

2

The Conditional Operator

```
if (n1 > n2)
   max = n1;
else
   max = n2;
```

can be written as

```
max = (n1 > n2) ? n1 : n2;
```

- The `?` and `:` together are call the *conditional operator* or *ternary operator.*

3

3

The `switch` Statement

- When a cascaded `if` statement has many possible outcomes, it can be hard to read.

```
int numBabies;
scanf("%d", &numBabies);

if (numBabies == 1)
      printf("Congratulations");
else if (numBabies == 2)
      printf("Twins!");
else if (numBabies == 3)
      printf("Wow, triplets!");
else if (numBabies == 4)
      printf("A miracle!");
else
      printf("You must be joking!!");
}
```

4

4

The `switch` Statement

- The `switch` statement is a multi-way branch
  - makes a decision based on a controlling expression
  - controlling expression evaluates to either an integer or char
- The `switch` statement
  - begins with the keyword `switch`
  - followed by the controlling expression in parentheses
  - then comes a list of cases, enclosed in braces where
    - each case consists of the keyword `case`
    - followed by: a constant called the *case label,* a colon and a list of statements

5

The `switch` Statement Syntax

```
switch (Expression)
{
   case value1:
        Statement(s);
        break;
   case value2:
        Statement(s);
        break;
   case valueN:
        Statement(s);
        break;
        ……
   default:
        Statement(s);
        break;
}
```

```
int numBabies;
scanf("%d", &numBabies);

switch (numBabies)
{
  case 1:
       printf("Congratulations");
       break;
  case 2:
       printf("Twins!");
       break;
  case 3:
       printf("Wow, triplets!");
       break;
  case 4:
       printf("A miracle!");
       break;
  default  :
       printf("You must be joking!!");
       break;
}
```

6

6

The `switch` Statement

- The case list is searched for a case label matching the controlling expression.
- If a matching case label is found, the statements associated with the matching case label is executed.
- If no match is found, the case labeled `default` is executed. The `default` case is optional, but recommended, even if it simply prints a message.

7

7

The `switch` Statement

- Repeated case labels are not allowed.
- The action for each case typically ends with the word `break`.
- The optional `break` statement prevents the execution of other cases.
- If no `break` statement is found for a particular case,
  - execution continues on with the next case
- Execution stops when
  - either a `break` statement is encountered or
  - the end of the `switch` statement is reached

8

8

The `switch` Statement

```
char grade = 'B';
switch (grade)
{
 case 'A':
      printf("Excellent!");
      break;
 case 'B':
 case 'C':
      printf("Well done");
      break;
 case 'D':
      printf("You passed");
      break;
 case 'F':
      printf("Better try again");
      break;
 default:
      printf("Invalid grade");
      break;
}
```

would print : Well done

9

9

The `switch` Statement

```
char grade = 'C';
switch (grade)
{
 case 'A':
      printf("Excellent!");
 case 'B':
 case 'C':
      printf("Well done");
 case 'D':
      printf("You passed");
 case 'F':
      printf("Better try again");
 default:
      printf("Invalid grade");
}
```

would print : Well done
You passed
Better try again
Invalid grade

- Common Mistake
  - Forgetting to include a break statement

10

10

Summary

- The Conditional Operator statement
- The `switch` statement
- Common mistakes