Programming Fundamentals
Lecture 2

Aamina Batool

Identifier

- Identifiers are **names of things** that appear in programs, such as variables, constants, and functions.

- Some identifiers are predefined; others are defined by the user.

Legal & Illegal Identifiers

- The following are legal identifiers in C++:
  - `first`
  - `conversion`
  - `payRate`

TABLE 2-1 Examples of Illegal Identifiers

Illegal Identifier	Description
`employee Salary`	There can be no space between `employee` and `Salary`.
`Hello!`	The exclamation mark cannot be used in an identifier.
`one+two`	The symbol + cannot be used in an identifier.
`2nd`	An identifier cannot begin with a digit.

Data Types

- Data Type: set of **values** together with a set of **operations** is called a data type

1

2

int Data Type

- Examples:
  - −6728
  - 0
  - 78
- Positive integers do not have to have a + sign in front of them
- No commas are used within an integer

`bool` Data Type

- `bool` type

  - Has two values, `true` and `false`

  - Manipulate logical (Boolean) expressions

- `true` and `false` are called logical values

- `bool`, `true`, and `false` are reserved words

Ⅲ- `char` Data Type

- The smallest integral data type

- Used for <u>characters</u>: letters, digits, and special symbols

- Each character is enclosed in single quotes

- Some of the values belonging to `char` data type are: `'A', 'a', '0', '*', '+', '$', '&'`

- A blank space is a character and is written `' '`, with a space left between the single quotes

Floating-Point Data Types

 C++ uses scientific notation to represent real numbers (floating-point notation)

TABLE 2-3 Examples of Real Numbers Printed in C++ Floating-Point Notation

Real Number	C++ Floating-Point Notation
75.924	7.592400E1
0.18	1.800000E−1
0.0000453	4.530000E−5
−1.482	−1.482000E0
7800.0	7.800000E3

Floating-Point Data Types (continued)

- **float**: represents any real number
  - Range: -3.4E+38 to 3.4E+38

- Memory allocated for the **float** type is 4 bytes

- **double**: represents any real number
  - Range: -1.7E+308 to 1.7E+308

- Memory allocated for **double** type is 8 bytes

Arithmetic Operators

- C++ Operators

  1   +   addition

  2   –   subtraction

  3   *   multiplication

  4   /   division

  5   %   remainder (mod operator)

- +, –, *, and / can be used with integral and floating-point data types

Order of Precedence

1-
- All operations inside of `()` are evaluated first

2-
- `*`, `/`, and `%` are at the same level of precedence and are evaluated next

3-
- `+` and `–` have the same level of precedence and are evaluated last

4-
- When operators are on the same level
    - Performed from left to right

Expressions

- If all operands are integers
  - Expression is called an integral expression
- If all operands are floating-point
  - Expression is called a floating-point expression
- An integral expression yields integral result
- A floating-point expression yields a floating-point result

- Mixed expression:

  - Has operands of different data types

  - Contains integers and floating-point

    5.4 * 2 – 13.6 + 18 / 2

Evaluating Mixed Expressions

- If operator has same types of operands
  - Evaluated according to the type of the operands
- If operator has both types of operands

1 — Integer is changed to floating-point

2 — Operator is evaluated

3 — Result is floating-point

Allocating Memory - Variable

- Variable: memory location whose content may change during execution

The syntax for declaring one variable or multiple variables is:

```
dataType identifier, identifier, . . .;
```

EXAMPLE 2-12

```
double amountDue;
int counter;
char ch;
int x, y;
string name;
```

Allocating Memory – Named Constant

- **Named Constant:** memory location whose content can't change during execution

- The syntax to declare a named constant is:

```
const dataType identifier = value;
```

- In C++, `const` is a reserved word

EXAMPLE 2-11

```
const double CONVERSION = 2.54;
const int NO_OF_STUDENTS = 20;
const char BLANK = ' ';
const double PAY_RATE = 15.75;
```

Assignment Statement

- The assignment statement takes the form:

  `variable = expression;`

- Expression is evaluated and its value is assigned to the variable on the left side

- In C++ `=` is called the assignment operator

- A C++ statement such as:

  `i = i + 2;`

  evaluates whatever is in `i`, adds two to it, and assigns the new value to the memory location `i`

Declaring & Initializing Variables

- Variables can be initialized when declared:

  ```
  int first=13, second=10;
  char ch=' ';
  double x=12.6, y=123.456;
  ```

- `first` and `second` are `int` variables with the values `13` and `10`, respectively

- `ch` is a `char` variable whose value is empty

- `x` and `y` are `double` variables with `12.6` and `123.456`, respectively

Different Operators

- **Arithmetic Operators (+ , -, /, %, *, =)**
- **Relational Operators (< , > , =, <=, >=, ==)**

Pseudocode

- **Pseudo code** is a kind of structured English for describing algorithms.

- It allows the designer to focus on the logic of the algorithm without being distracted by details of language syntax.

- At the same time, the pseudo code needs to be complete. It describes the entire logic of the algorithm so that implementation becomes a rote mechanical task of translating line by line into source code.

How to write a Pseudocode

1. A computer can receive information
   - Read (information from a file)
   - Get (information from the keyboard)
2. A computer can put out information
   - Write (information to a file)
   - Display (information to the screen)

How to write a Pseudocode

3. A computer can perform arithmetic
   - Use actual mathematical symbols or the words for the symbols
   - Add number to total
   - Total = total + number
   - +, -, *, /
   - Calculate, Compute also used

How to write a Pseudocode

4. A computer can assign a value to a piece of data
   - 3 cases
     1. To give data an initial value,
        - Initialize, Set
     2. To assign a value as a result of some processing ,
        - '=',
        - x = 5 + y
     3. To keep a piece of information for later use,
        - Save, Store
   - For above three cases, data must be declared through "Declare"

How to write a Pseudocode

5. A computer can compare two piece of information and select one of two alternative actions

- IF (condition)

  - some action

- ELSE

  - alternative action

6. A computer can repeat a group of actions

Sequence

- Execution of one step after another. This is represented as a sequence of pseudo code statements:

- Statement 1

- Statement 2

- Statement 3

- Example:

  - Read three numbers

  - Add three numbers

  - Display total of three numbers

Selection

- Presentation of a condition and the choice between two actions, the choice depending on whether the condition is true or false. This construct represents the decision making abilities of the computer to compare two pieces of information and select one of two alternative actions. In pseudo code, selection is represented by the keywords IF, THEN, ELSE and ENDIF

- IF condition p is true

  - statement(s) in true case

- ELSE

  - statement(s) in false case

Selection

- Example:
  - IF (student is part_time)
    - Add one to part_time_count
  - ELSE
    - Add one to full_time_count

- A variation – We don't need the ELSE structure – The null ELSE
- IF condition p is true
  - statement(s) in true case

Examples

- Write pseudo code that reads two numbers and multiplies them together and print out their product.

  - Declare num1, num2, product

  - Input num1 , num2

  - Set product to num1*num2

  - Print product

Examples

- Write pseudo code that tells a user that the number they entered is not a 5 or a 6.
- Solution # 1:
  - Declare num
  - Input num
  - If(num = 5)
    - Print "your number is 5"
  - Else if (num = 6)
    - Print "your number is 6"
  - Else
    - Print "your number is not 5 or 6"

Examples

- Solution # 2:
  - Input num
  - If(num = 5 or num = 6)
    - Print "your number is a 5 or a 6"
  - Else
    - Print "your number is not 5 or 6"

Examples

- Solution # 3:
  - Input num
  - If(num is not 5 and num is not 6)
    - Print "your number is not 5 or 6"
  - Else
    - Print "your number is a 5 or a 6"

References

1. C++ Programming: From Problem Analysis to Program Design, Third Edition
2. https://www.just.edu.jo/~yahya-t/cs115/