C++ Basics

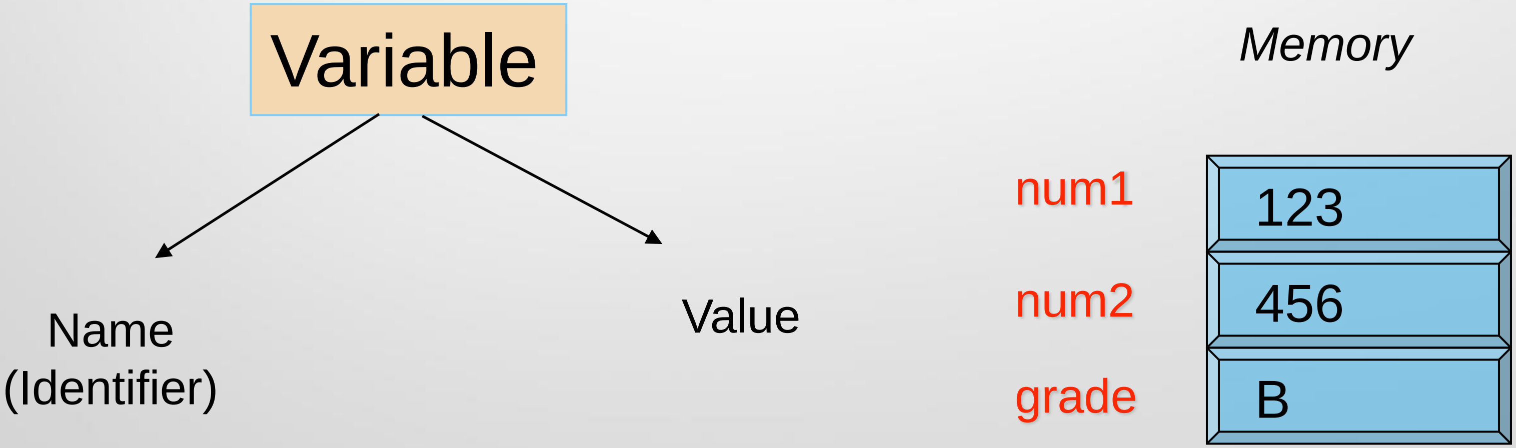
Lecture 2

Description

In this chapter we will discuss about the variables and how it is used in our programs and how to declare the variable with a simple understanding of the memory storage. The input and output statement is also discussed

Variables and Assignments

- Programs manipulate data such as numbers and letters
- C++ uses variables to name & store data



Variable

Name
(Identifier)

Value

Memory

num1 123

num2 456

grade B

Variables and Assignments

- Variable Declarations
  - Syntax:

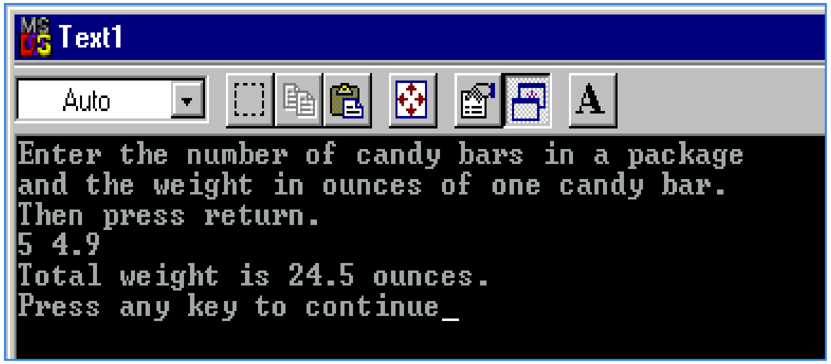
    Type_Name Variable_Name1, Variable_Name_2, …;
  - Example:

    int count, total;

    double price;
- All variables must be declared before they are used in the program.

Variables and Assignments

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

int main()
{  int  number_of_bars;
   double  one_weight,  total_weight;

   cout<< "Enter the number of candy bars in a package\n";
   cout<< "and the weight in ounces of one candy bar.\n";
   cout<< "Then press return.\n";

   cin>> number_of_bars;
   cin>> one_weight;


   total_weight = one_weight * number_of_bars;

   cout<< "Total weight is " <<total_weight << " ounces.\n";

   return 0;
}
```



```
Enter the number of candy bars in a package
and the weight in ounces of one candy bar.
Then press return.
5 4.9
Total weight is 24.5 ounces.
Press any key to continue_
```

Variables and Assignments

- Names: Identifiers
  - Begin with a letter or underscore
  - Remaining characters must be
    - Letters or
    - Digits or
    - Underscore

sum	✓
Big_Bonus	✓
3X	✗
_address	✓
%change	✗
program1.cpp	✗
_3X	✓
price-1	✗
total5*	✗

Variables and Assignments

- Notes on Identifiers
  - C++ is case sensitive
    - Average
    - AVERAGE
    - average
  - Use meaningful names
  - Keywords/reserved words
    - int
    - double

Variables and Assignments

Assignment Statements

- Syntax:

  Variable = *Expression* ;

- Examples:

  distance = speed_rate * time;

  count = count + 2;

  weight = 35;

Variables and Assignments

- Initializing variables
  - Syntax:

```
Type     Variable_Name_1 = Expression_for_value_1,
         Variable_Name_2 = Expression_for_value_2,..;

Type     Variable_Name_1 ( Expression_for_value_1),
         Variable_Name_2 ( Expression_for_value_2),..;
```

  - Examples:
    - int count = 0, max = 555;
    - int   count(0), max(555);

Input and Output

- Input stream
  - The stream of input that is being fed into the computer for the program to use
  - cin (cin>> number_of_bars;)
- Output stream
  - The stream of output generated by the program
  - cout (cout<< "Enter the number of candy bars.\n";)

Input and Output

Input Using cin

- Syntax:

  cin >> *Variable_1* >> *Variable_2* >>...;

- Examples:
  - cin >> number >> size;
  - cin >> grade1

    >> grade2;

- cin >> grade1;

  cin >> grade2;

Input and Output

Output Using cout

- Syntax:

  cout << *Variable_or_string_1* << *Variable_or_string_2* << … ;

- Examples:
  - cout << number << size;
  - cout << “Hello \n”;

Input and Output

- Include directive
- Using directive
- Namespaces (collection of names)

```
# include <iostream>

using namespace std;
```

Input and Output

- Escape Sequences

  The backslash \ preceding a character tells the compiler that the sequence following the \ doesn’t have the same meaning as the character appearing by itself.

- New_line \n
- Horizontal tab \t
- Alert \a
- Backslash \\
- Double quote\”
- Others: v, b, r, ?, :, \000, \xhhh

Input and Output

- New line & Blank lines
  - cout<< “\n”;
  - cout<< endl;
- If you could include the \n at the end of a longer string, then use \n.
- If the \n would appear by itself as the short string “\n”, then use endl instead.

Input and Output

- Formatting numbers with a decimal point

```
double price = 84.50;

cout << “The price is $” << price<<endl;
```

```
The price is $84.5
```

```
The price is $84.500000
```

```
The price is $84.50
```

```
The price is $84.5000e01
```

Input and Output

- Magic Formula

```
cout.setf (ios::fixed);
 cout.setf(ios::showpoint);
 cout.precision(2);
```

Input and Output

- Line Breaks in I/O

  You can keep input and output on the same line by omitting the \n or endl at the end of the last prompt line.

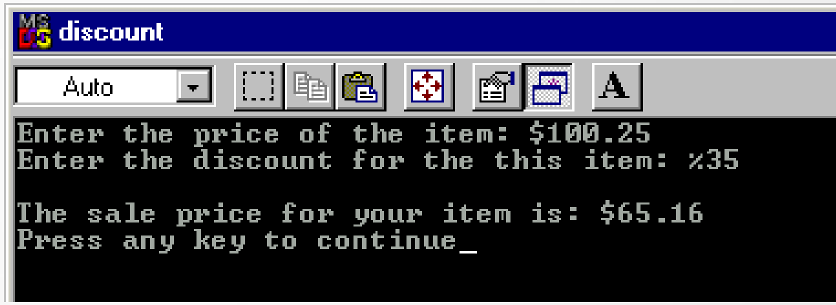
- Example:

```
cout<< "Enter the cost per person: $";
cin   >> cost_per_person;
```

```
Enter the cost per person: $5.40
```

Lab Exercise

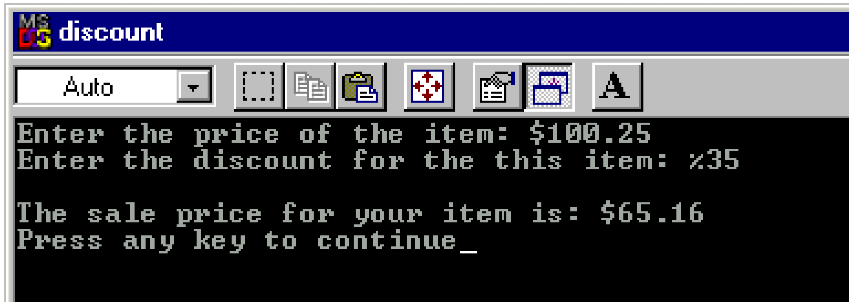
- Your local department store is having its annual sale. Write a program that calculates the sale price for items in the store. The program should prompt the user for the original price and the discount (10%, 25%, etc.)



```
Enter the price of the item: $100.25
Enter the discount for the this item: %35

The sale price for your item is: $65.16
Press any key to continue_
```

Lab Exercise

```
include <iostream>

using namespace std;

int main()

{         double discount, price;

          cout<<"Enter the price of the item: $";

          cin>> price;

          cout<<"Enter the discount for the this item: %";

          cin>> discount;

          price = price - (price * discount/100);

          cout.setf (ios::fixed);

          cout.setf(ios::showpoint);

          cout.precision(2);

          cout<<"\nThe sale price for your item is: $"<< price;

          cout<<endl;

          return 0;

}
```



```
Enter the price of the item: $100.25
Enter the discount for the this item: %35

The sale price for your item is: $65.16
Press any key to continue_
```

Lab Exercise

1. Write a Program to convert a temperature in degrees Fahrenheit to degree Celcius.

-
- Data Requirement
- Problem input
- int Fahrenheit
- Problem OutPut
- Float Celcius
- Formula
- Celcius = (5/9) * (faherenheit – 32)

Lab Exercise

1. write a Program to read two data items and print their sum, difference, product, and quotient.
2. Write a program that reads in the length and width of a rectangular yard and the length and width of a rectangular house situated in the yard. Your program should compute the time required to cut the grass at the rate of 2 square meters per second

Summary

- Variables stores data
- C++ uses variable to name and store data
- Rules to declare variables
- Assigning the values
- Input and output statements in c++

Thank You