



3

principles of programming

Data types & variables

Department of Cybersecurity

Exercise 3.2

- Write a C++ program to calculate and display total amount of the given unit price and quantity of an item.



Data types

Data type selection



Memory allocation

| | | | |
|-----------------------|-----------------------|-----------------------|-----------------------|
| int a (4 bytes) | | | |
| float f (4 bytes) | | | |
| char c1 (1 byte) | Unused | | |
| char c2 (1 byte) | Unused | | |
| char d[0] (1 byte) | char d[1] (1 byte) | char d[2] (1 byte) | char d[3] (1 byte) |
| double g (8 bytes) | | | |

Data Types

- Use various variables to store various information
- **Variables are reserved memory locations to store values**
- When you create a variable you reserve some space in memory
- **Based on the data type of a variable, the operating system allocates memory and decides what can be stored in the reserved memory**

C++ Data Types

- basic C++ data types

| Type | Keyword |
|-----------------------|---------|
| Boolean | bool |
| Character | char |
| Integer | int |
| Floating point | float |
| Double floating point | double |

Modifiers

- Several of the basic types can be modified using one or more of these type modifiers
 - signed
 - unsigned
 - short
 - long

Character types

char

- They can represent a single character, such as 'A' or '\$'. The most basic type is char, which is a one-byte character.

| Type | Typical Bit Width | Typical Range |
|---------------|-------------------|-------------------------|
| char | 1byte | -127 to 127 or 0 to 255 |
| unsigned char | 1byte | 0 to 255 |
| signed char | 1byte | -127 to 127 |

ASCII Code for the characters

| | | | | | | | | | | | | | | | |
|----|------|----|---|----|------|----|-----|----|-----|----|-----|-----|-----|-----|-----|
| 0 | '\0' | 16 | — | 32 | ' ' | 48 | '0' | 64 | @ | 80 | 'P' | 96 | '`' | 112 | 'p' |
| 1 | — | 17 | — | 33 | '!' | 49 | '1' | 65 | 'A' | 81 | 'Q' | 97 | 'a' | 113 | 'q' |
| 2 | — | 18 | — | 34 | '"' | 50 | '2' | 66 | 'B' | 82 | 'R' | 98 | 'b' | 114 | 'r' |
| 3 | — | 19 | — | 35 | '#' | 51 | '3' | 67 | 'C' | 83 | 'S' | 99 | 'c' | 115 | 's' |
| 4 | — | 20 | — | 36 | '\$' | 52 | '4' | 68 | 'D' | 84 | 'T' | 100 | 'd' | 116 | 't' |
| 5 | — | 21 | — | 37 | '%' | 53 | '5' | 69 | 'E' | 85 | 'U' | 101 | 'e' | 117 | 'u' |
| 6 | — | 22 | — | 38 | '&' | 54 | '6' | 70 | 'F' | 86 | 'V' | 102 | 'f' | 118 | 'v' |
| 7 | '\a' | 23 | — | 39 | '\'' | 55 | '7' | 71 | 'G' | 87 | 'W' | 103 | 'g' | 119 | 'w' |
| 8 | '\b' | 24 | — | 40 | '(' | 56 | '8' | 72 | 'H' | 88 | 'X' | 104 | 'h' | 120 | 'x' |
| 9 | '\t' | 25 | — | 41 | ')' | 57 | '9' | 73 | 'I' | 89 | 'Y' | 105 | 'i' | 121 | 'y' |
| 10 | '\n' | 26 | — | 42 | '*' | 58 | ':' | 74 | 'J' | 90 | 'Z' | 106 | 'j' | 122 | 'z' |
| 11 | — | 27 | — | 43 | '+' | 59 | ';' | 75 | 'K' | 91 | '[' | 107 | 'k' | 123 | '{' |
| 12 | '\f' | 28 | — | 44 | ',' | 60 | '<' | 76 | 'L' | 92 | '\' | 108 | 'l' | 124 | ' ' |
| 13 | '\r' | 29 | — | 45 | '-' | 61 | '=' | 77 | 'M' | 93 | ']' | 109 | 'm' | 125 | '}' |
| 14 | — | 30 | — | 46 | '.' | 62 | '>' | 78 | 'N' | 94 | '^' | 110 | 'n' | 126 | '~' |
| 15 | — | 31 | — | 47 | '/' | 63 | '?' | 79 | 'O' | 95 | '_' | 111 | 'o' | 127 | — |

Numerical integer types

- They can store a number value, such as 7 or 1024

| Data Type | Size | Range |
|----------------|---------|--|
| short | 2 bytes | -32,768 to +32,767 |
| unsigned short | 2 bytes | 0 to +65,535 |
| int | 4 bytes | -2,147,483,648 to +2,147,483,647 |
| unsigned int | 4 bytes | 0 to +4,294,967,295 |
| long | 8 bytes | -9,223,372,036,854,775,808 to +9,223,372,036,854,775,807 |
| unsigned long | 8 bytes | 0 to +18,446,744,073,709,551,615 |

Floating-point types

- They can represent real values, such as 3.14 or 0.01

| Type | Size in Bytes* | Minimum Positive Value* | Maximum Positive Value* |
|-------------|----------------|-------------------------|-------------------------|
| float | 4 | 3.4E-38 | 3.4E+38 |
| double | 8 | 1.7E-308 | 1.7E+308 |
| long double | 10 | 3.4E-4932 | 1.1E+4932 |

The **E** means exponent of base-10.

Instead of writing 3.504×10^{12} , in C++ we write **3.504E12**.

Size of a data type

- Use **sizeof()** function to check the size

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

int main()
{
    cout << "Size of char : " << sizeof(char) << endl;
    cout << "Size of int : " << sizeof(int) << endl;
    cout << "Size of short int : " << sizeof(short int) << endl;
    cout << "Size of long int : " << sizeof(long int) << endl;
    cout << "Size of float : " << sizeof(float) << endl;
    cout << "Size of double : " << sizeof(double) << endl;
    cout << "Size of wchar_t : " << sizeof(wchar_t) << endl;
    return 0;
}
```

Answer

- Result which can vary from machine to machine/
Compilers

```
Size of char : 1  
Size of int : 4  
Size of short int : 2  
Size of long int : 4  
Size of float : 4  
Size of double : 8
```

Define Variables in C++

- Tell the compiler where and how much to create the storage for the variable
- Syntax
 - Data Type <space> Variable Name;
 - int number ;
 - char letter ;
 - float emp_salary ;

Variable name (rule)

- Can use only letters, digits and underscore
- The first character must be a letter or underscore
- Case sensitive
- Cannot use keywords
- No limits on length

C++ Variables

- Better not to begin a variable name with underscore.
- To form a name from two or more words, separate them with underscore
- Example
 - `int student_age;`
 - `float employe_salary ;`
 - `char grade ;`

Local & Global Variables

```
#include <iostream>
```

```
using namespace std;
```

```
int data_size =250;
```

```
int main()
```

```
{
```

```
    int age;
```

```
    return 0;
```

```
}
```

Global variable

data_size

age

Local variable

Variables in a program

```
#include <iostream>
using namespace std;
int main()
{
    int age;
    float salary;
    age =23;
    salary = 25000.00;

    cout << age;
    cout << salary;

    return 0;
}
```

Exercise 3.3

- Write a C++ program to calculate and display total amount of the given unit price and quantity of an item.

Answer

```
#include <iostream>
using namespace std;
int main()
{
    int item_quantity;
    float unit_price;
    float total_salary;

    item_quantity = 5;
    unit_price = 250;

    total_salary = unit_price * item_quantity;

    cout << "-----\n";
    cout << " Unit price      " << unit_price << endl;
    cout << " Item quantity  " << item_quantity << endl;
    cout << " Total salary   " << total_salary << endl;
    cout << "-----";
    return 0;
}
```

Exercise 3.1

- Write a C++ program to display the following output.

```
-----  
Unit price   : 250.00  
Quantity     : 5  
-----  
Total        :1250.00  
=====
```

Constant variables

- Constants are declared like variables with the addition of the **const** keyword

const double PI =3.14159 ;

Once declared and initialized, a constant can be used like a variable

- A constant may not be reassigned

C++ Memory concept

- Variable names correspond to location in the computer's memory
- Every variable has a name, a type, a size and a value
- A memory cell is never empty. But its initial contents may be meaningless to your program.

Garbage values

```
char ch;  
int intv;  
float floatv;  
long longv;  
double doublev;  
bool boolv;
```

```
cout << "char value : "<< ch << endl;  
cout << "int value : "<< intv << endl;  
cout << "float value : "<< floatv << endl;  
cout << "long value : "<< longv << endl;  
cout << "double value : "<< doublev << endl;  
cout << "bool value : "<< boolv << endl;
```

```
char value :  
int value : 6946708  
float value : 6.00597e-039  
long value : 1980658781  
double value : 4.68194e+260  
bool value : 249
```



Input / Output

Standard Streams

Input

- **cin** is the standard input, normally the keyboard.
- To input data from keyboard use the word '**cin**', followed by the 'Extraction' operator(>>)

```
cin >> x;
```

- Wait for a value to be entered at the keyboard and (when enter is pressed) will put that value into variable ' x '

Exercise 3.4

- Write a C++ program to read two numbers from keyboard and display the total.

Answer

```
#include <iostream>
using namespace std;
int main()
{
    int num1, num2, total;

    cout << "Enter value 1: ";
    cin >> num1;
    cout << "Enter value 2: ";
    cin >> num2;

    total = num1 + num2;

    cout << "Total is : " << total << endl;
    return 0;
}
```

Exercise 3.5

Create a C++ program to calculate and display total amount of given unit price and quantity of the some item.

Answer

```
int main()
{
    float unit_price;
    float total;
    int quantity;

    cout << "Enter Unit price:";
    cin >> unit_price;
    cout << "Enter Quantity :";
    cin >> quantity;
    total = unit_price * quantity;
    cout << "Total is "<<total<<endl;
    return 0;
}
```

Input

```
int age;  
float salary;  
char gender;  
// input separately  
cout << "Enter age, salary and gender\n";  
cin >> age;  
cin >> salary;  
cin >> gender;  
// input at once  
cout << "Enter age, salary and gender\n";  
cin>> age >> salary >> gender;  
cout << "OK";
```


Output

- To output data onto the screen, use the word **'cout'**, followed by the 'insertion' operator (<<).

```
cout << "this is a String";  
cout << "Value " << total;
```

Exercise 3.6

Write a C++ program to read price of the 3 items and print the total

Answer

```
float item1, item2, item3, total;
cout << "Enter item 1 : ";
cin >> item1;
cout << "Enter item 2 : ";
cin >> item2;
cout << "Enter item 3 : ";
cin >> item3;
cout.setf(ios::fixed);
cout.precision(2);
cout << "-----" << endl;
cout << "Item 1";
cout.width(10);
cout << item1 << endl;
cout << "Item 2";
cout.width(10);
cout << item2 << endl;
cout << "Item 3";
cout.width(10);
cout << item3 << endl;
cout << "-----" << endl;
```



```
Enter item 1 : 1200
Enter item 2 : 23.456
Enter item 3 : 15000.00
-----
Item 1      1200.00
Item 2       23.46
Item 3     15000.00
-----
```

Examples

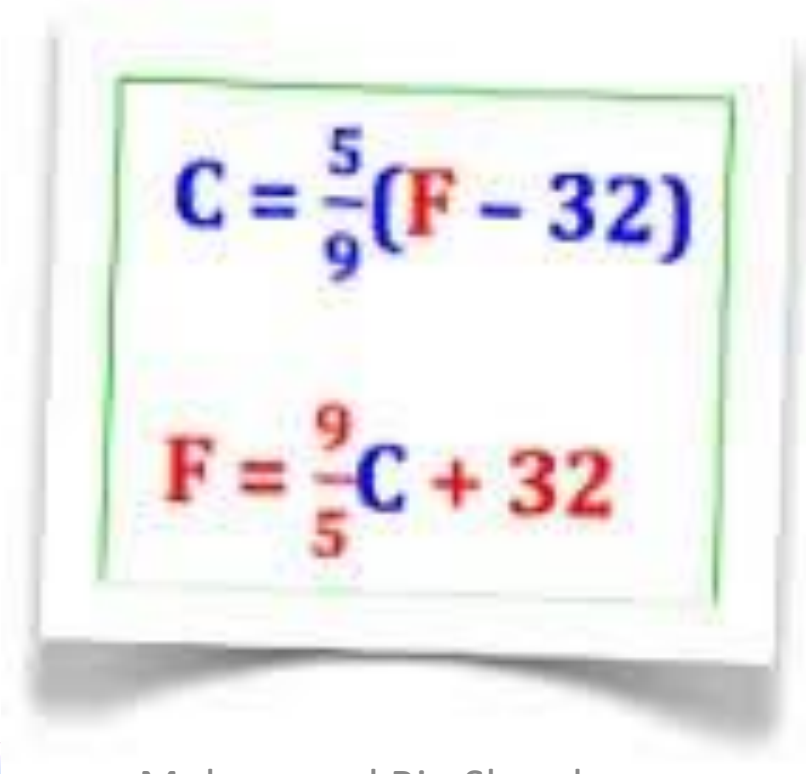
1. Write a C++ program to read 3 integer numbers and find the total and average
2. Write a C++ program which will convert a weight in KG to pounds .

1 Kg = 2.2046 pounds



Example

- Write a C++ program which will convert a Celsius temperature into Fahrenheit


$$C = \frac{5}{9}(F - 32)$$
$$F = \frac{9}{5}C + 32$$

Summary

- Data types (char, int, float, long, double, boo)
- Variables (local , global and constant)
- Input fro keyboard cin >>
- Output cout <<

