Students:
Section 2.23 is a part of 2 assignments: CSC108 CH02.11-2.24 C2B

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.

Includes: CA
Due: 02/06/2025, 11:59 PM EST

2.23 Auto (since C++11)

Using the auto specifier

The keyword **auto** tells the compiler to determine the variable's type using the initial value given. The auto specifier has been supported in all versions of C++ since the release of C++11. Using auto can make variable declaration easier for the programmer when working with complicated data types.

Table 2.23.1: Basic types found by the compiler given the initial value.

Code	Variable	Type found
`auto v = 2;`	v	int
`auto w = 0.5;`	w	double
`const auto x = 7;`	x	const int
`auto y = 'h';`	y	char
`auto z = "apple";`	z	const char *

Feedback?

*Above, "apple" is a string literal. When an auto variable is initialized with a string literal, the compiler interprets the type as a const char *, not as a string. const char * means a constant pointer to a character array, and is a simple data type described in detail elsewhere.*

PARTICIPATION ACTIVITY 2.23.1: Auto in variable declarations.

What is x's type?

1) `auto x = -9;`
   - int
   - double
   - Error
2) `auto x = 0.01;`
   - int
   - double
   - Error
3) `const auto x = '5';`
   - int
   - char
   - Error
4) `auto x = "Hello";`
   - char
   - string
   - const char *
5) `auto x;`
   - int
   - double
   - Error
6) `int v = 1;`
   `auto x = v;`
   - int
   - v
   - Error
7) `auto x = 1;`
   `x = 2.0;`
   - double
   - int
   - Error

Feedback?

Printing the type of an auto variable

During debugging, a programmer may want to see if the compiler has correctly determined an auto variable's type. The ***typeid*** operator reports a variable's type. typeid's name() function returns a string describing the variable's type. The type description is implementation-dependent. The g++ compiler uses "d" for double (or const double), "c" for char (or const char), and "i" for integer (or const int). Ex: If variable x is of type int or const int, then `typeid(x).name()` returns "i".

PARTICIPATION ACTIVITY 2.23.2: typeid reports the type of an auto variable.

Captions
1. The compiler sees that 4.5 is the initial value of x, so the compiler determines x to be a double.
2. The compiler initializes x with value 4.5. x's value may be changed later in the program, but x's type does not change once determined.
3. When the program runs, typeid reports x's type. typeid(x).name() returns "d" for double, and "d" is output to the screen.

Feedback?

zyDE 2.23.1: Explore the information returned by typeid.

Change the initial value of variable x and observe the type determined for x by the compiler.

Load default template... Run

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

int main() {
   auto x = 1;

   cout << typeid(x).name() << endl;

   return 0;
}
```

Feedback?

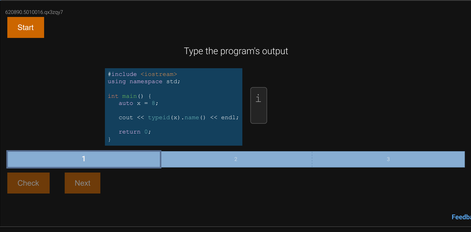
PARTICIPATION ACTIVITY 2.23.3: More about typeid.

Assume the g++ compiler is used.

1) `typeid(x).name();` returns "d". What is x's type?
   - int
   - double
   - decimal
2) What is the output?
   ```
   auto x = 1;
   x = 2.0;
   cout << typeid(x).name() << endl;
   ```
   - i
   - d
   - Error
3) What is the output?
   ```
   auto x = "hello";
   cout << typeid(x).name() << endl;
   ```
   - c
   - PKc
   - string

Feedback?

CHALLENGE ACTIVITY 2.23.1: Enter the output returned by typeid compiled with g++.



Feedback?

Exploring further:

- CppReference.com (auto)
- MSDN C++ reference (auto)
- CppReference.com (typeid)
- MSDN C++ reference (typeid)

How was this section? | Provide section feedback

Activity summary for assignment: CSC108 CH02.11-2.24 C2B — 51 / 51 points

Due: 02/06/2025, 11:59 PM EST — 51 / 51 points submitted to BlackboardLearn

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