15CSE102
Computer Programming

Credits: jobssearch24.com

Know Your Variables

Know Your Variables

Variables must have a name

But not these names!!

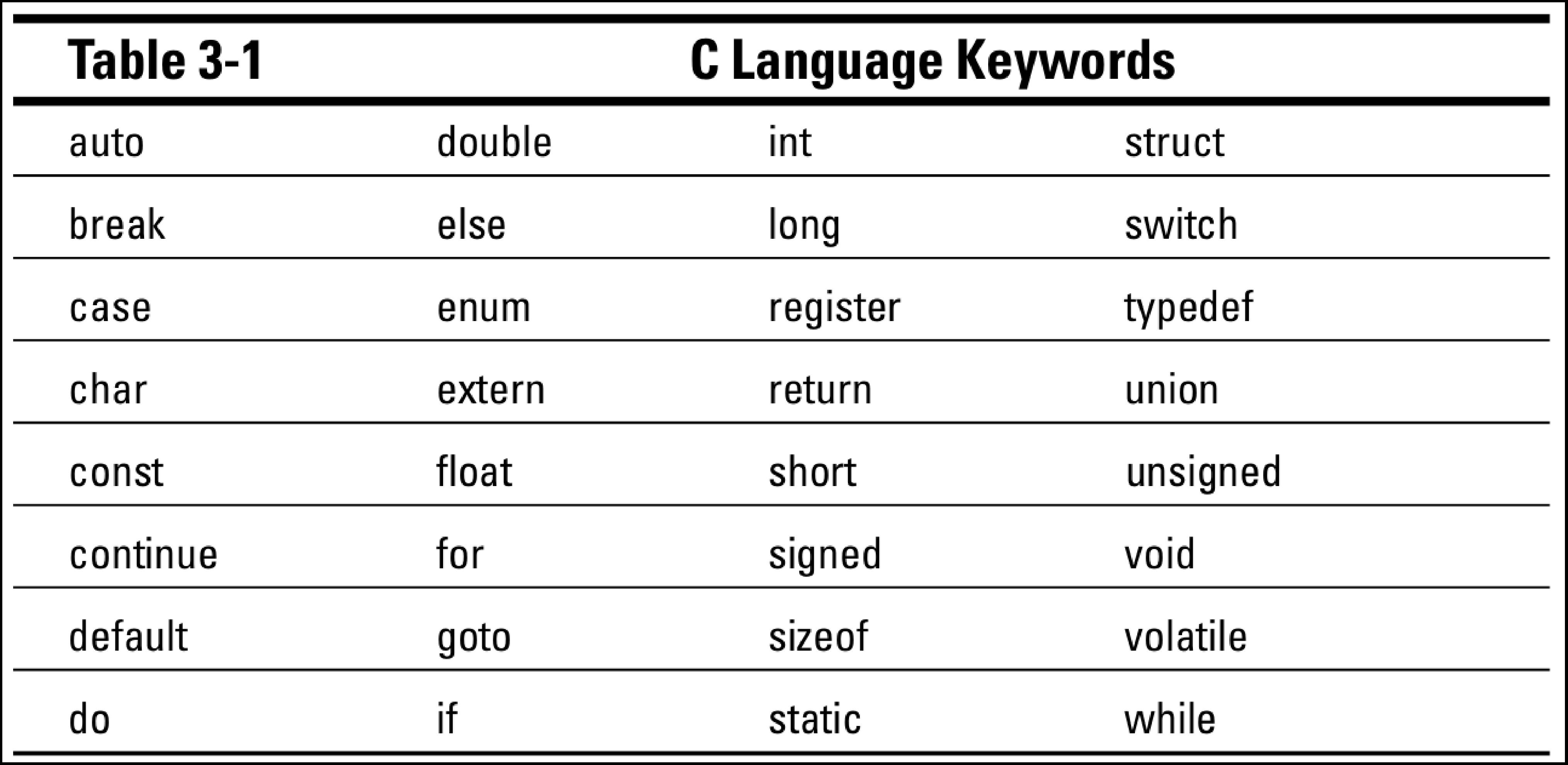


Table 3-1 **C Language Keywords**

auto	double	int	struct
break	else	long	switch
case	enum	register	typedef
char	extern	return	union
const	float	short	unsigned
continue	for	signed	void
default	goto	sizeof	volatile
do	if	static	while

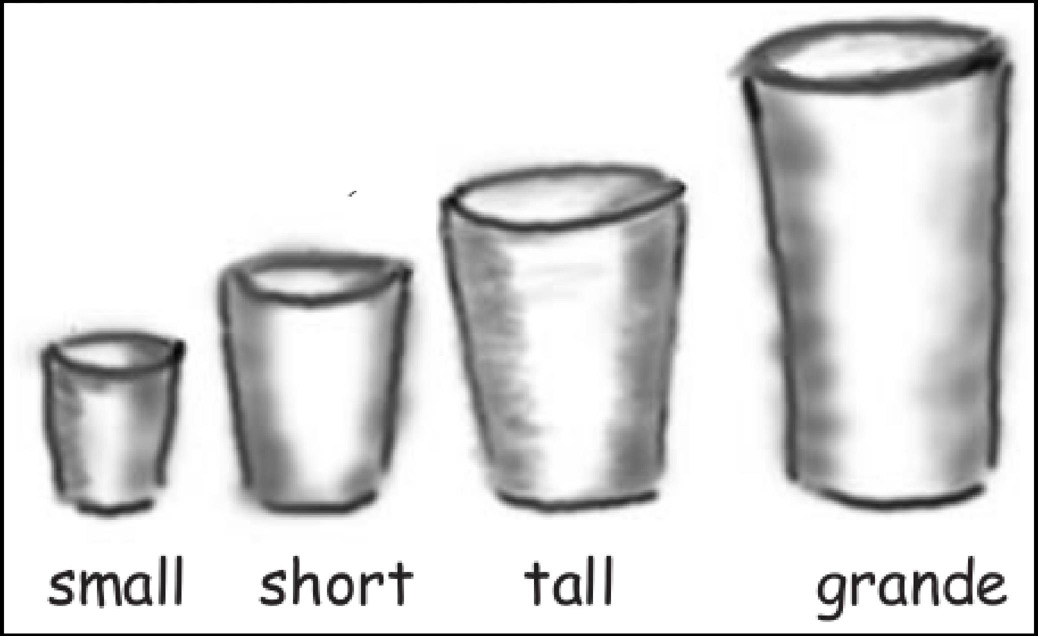
Credits: C for Dummies

Know Your Variables

Variables must have a name

Variables must have a type

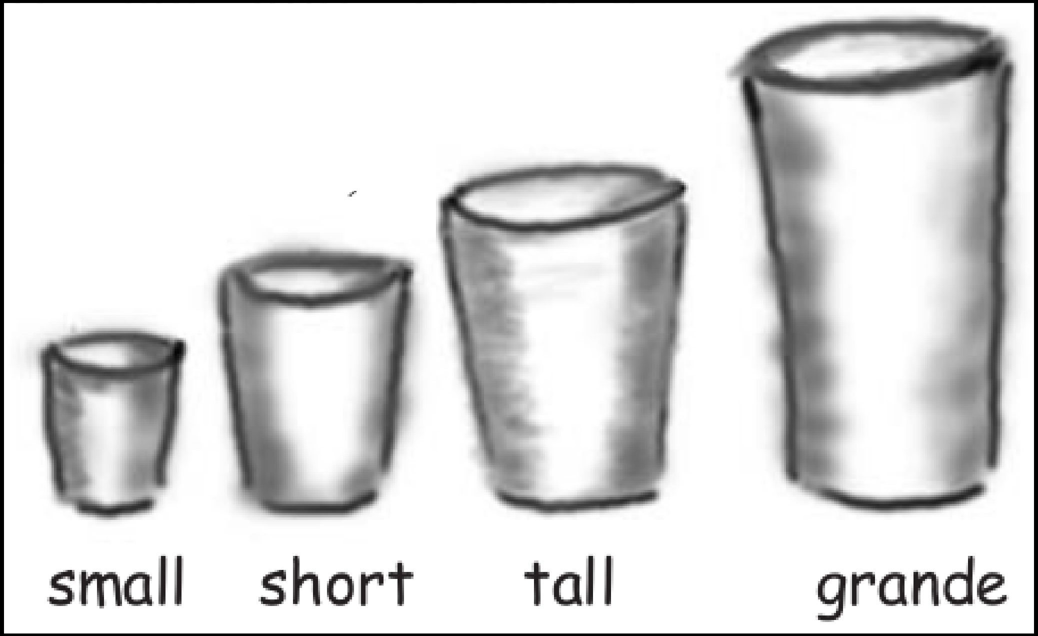
Variable Types



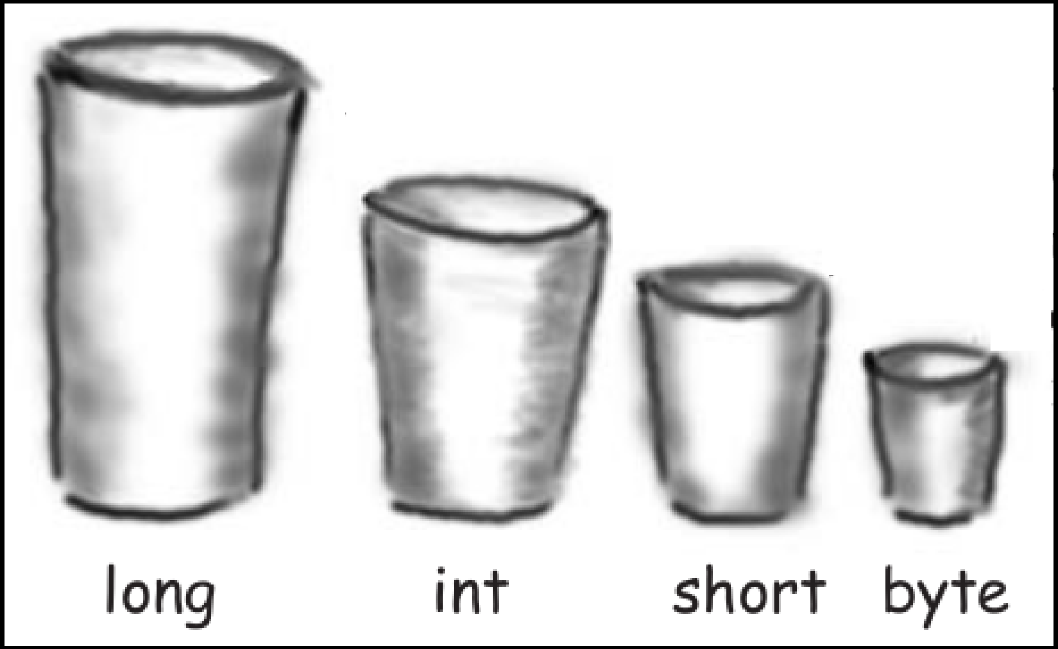
Variables are like cups/containers that contain something!

Credits: Head First Java

Variable Types



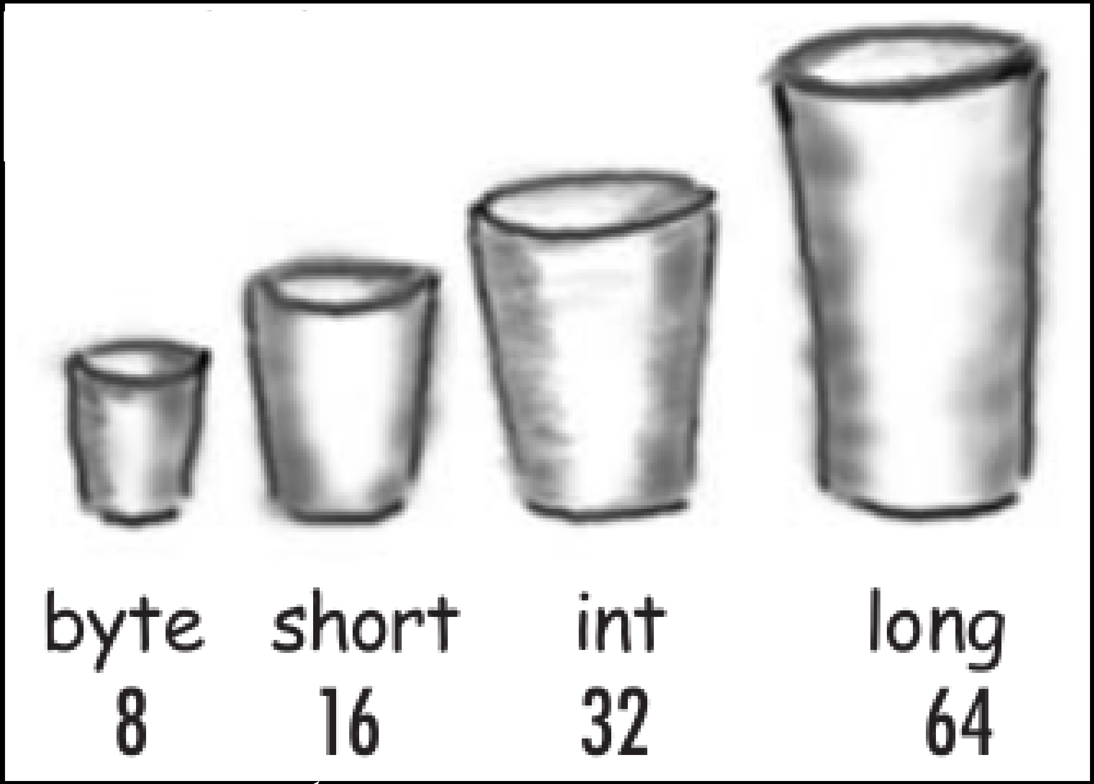
Variables are like cups/containers that contain something!



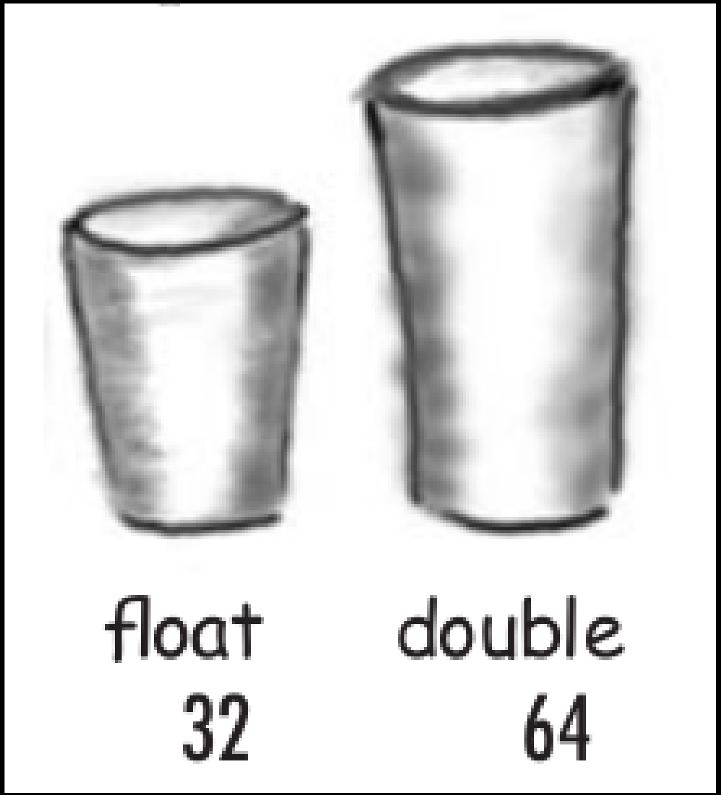
Integers in different flavors!!

Credits: Head First Java

Numeric Primitives



char variable type stores characters



Credits: Head First Java

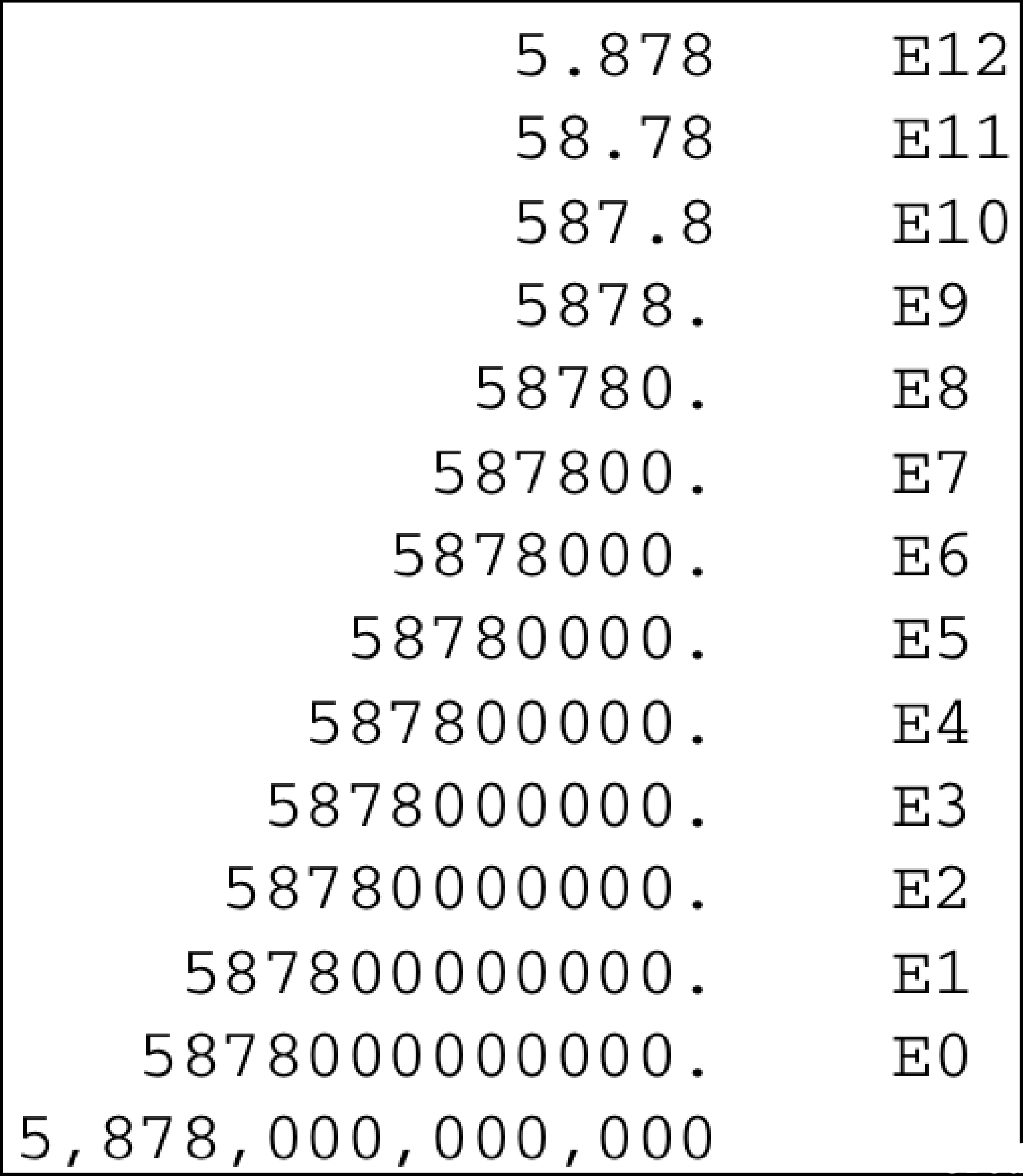
Ranges

Integer Type	No of Bytes	Minimum value that can be stored	Maximum Value that can be stored
short int	2	-32,768	32,767
int	4	-2,147,483,648	2,147,483,647
long int	4	-2,147,483,648	2,147,483,647
long long int	8	-9,223,372,036,854,775,807	9,223,372,036,854,775,806

Ranges

Integer Type	No of Bytes	Minimum value that can be stored	Maximum Value that can be stored
float	4	3.4 e -38	3.4 e 38
double	8	1.7 e -308	1.7 e 308
long double	12	3.4 e - 4932	1e+493

Scientific Notation



5.878	E12
58.78	E11
587.8	E10
5878.	E9
58780.	E8
587800.	E7
5878000.	E6
58780000.	E5
587800000.	E4
5878000000.	E3
58780000000.	E2
587800000000.	E1
5878000000000.	E0
5,878,000,000,000	

Credits: C For Dummies

Know Your Variables

Variables must have a name

Variables must have a type

Know Your Variables

Variables must have a name

Variables must have a type

Variables must be declared before their usage

$$\underset{\underset{\text{type}}{\nearrow}}{\texttt{int}}\ \underset{\underset{\text{name}}{\nwarrow}}{\texttt{count;}}$$

Declaration & Assignment

```
int x; // variable declarations
int y;
int z;
```

Declaration & Assignment

```
int x; // variable declarations
int y;
int z;
// ways to assign values to variables
x = 12;// direct assignment of literal
       // value to variable
```

Declaration & Assignment

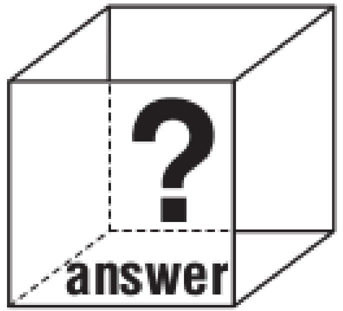
```
int x; // variable declarations
int y;
int z;
// ways to assign values to variables
x = 12;// direct assignment of literal
       // value to variable
y = z; // assign value of one variable
       // to another variable
```

Declaration & Assignment

```
int x; // variable declarations
int y;
int z;
// ways to assign values to variables
x = 12;// direct assignment of literal
       // value to variable
y = z; // assign value of one variable
       // to another variable
z = x + 43; // thru an expression
```

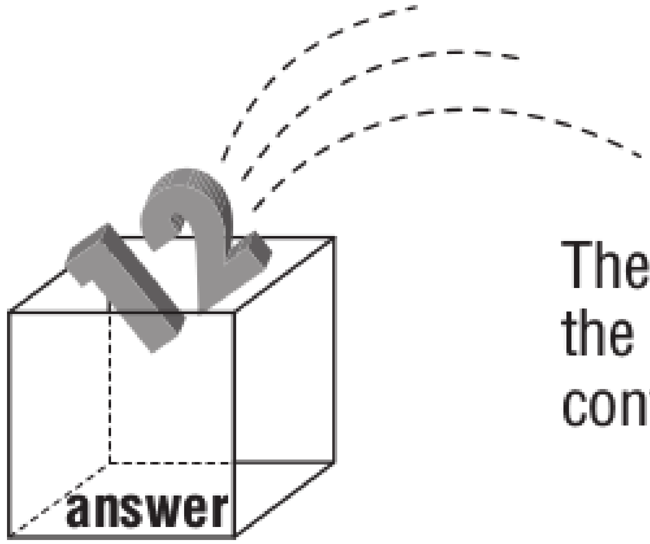
Declaration & Assignment

A `int answer;`



The variable `answer` has not been assigned a value. So we put a "?" in it to indicate that it's in an unknown state.

B `answer = (1+2) * 4;`



The variable `answer` is assigned the value of the expression `(1+2)*4`. The box is shown containing the value 12.

Credits: Practical C Programming

Printing Variable Value

```
// Did you notice! Simultaneous
// declaration and assignment

// First assignment can also be called
// variable initialization
int height = 6;
```

Printing Variable Value

```
int height = 6;
// f stands for formatted
// output can be formatted
printf("Height: %d\n", height);
```

What are these?

Printing Variable Value

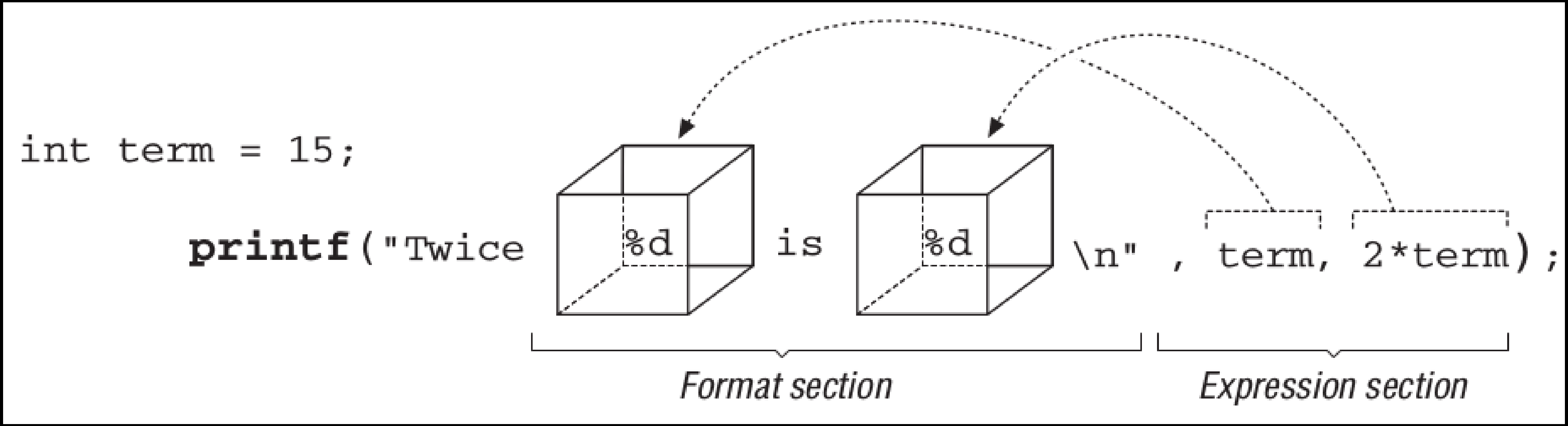
```c
int height = 6;
// f stands for formatted
// output can be formatted
printf("Height: %d\n", height);
```

A place holder

It is called a format specifier

d stands for decimal

Printf Structure



```
int term = 15;

      printf("Twice %d is %d\n" , term, 2*term);
```

Credits: Practical C Programming

Printing Variable Value

```
int height = 6;
// f stands for formatted
// output can be formatted
printf("Height: %d\n", height);
```

Escape sequence
n stands for newline

Printf Structure

%i or %d	int
%c	char
%f	float
%lf	double
%s	string

Escape sequence	Description
\n	Newline. Position the cursor at the beginning of the next line.
\t	Horizontal tab. Move the cursor to the next tab stop.
\a	Alert. Produces a sound or visible alert without changing the current cursor position.
\\	Backslash. Insert a backslash character in a string.
\"	Double quote. Insert a double-quote character in a string.

Credits: c.camden.rutgers.edu; Deitel & Deitel

Reading User Input

Ofcourse in a variable

```
// The following declaration allocates
// memory enough to accommodate
// an integer and names that location
// as height
int height;
```

Reading User Input

Ofcourse in a variable

```
int height;
// while printf is intended for output
// scanf is intended for accepting inputs
scanf("%d", &height);
```

Reading User Input

Ofcourse in a variable

```
int height;
// while printf is intended for output
// scanf is intended for accepting inputs
scanf("%d", &height);
```

What is this?

Reading User Input

Ofcourse in a variable

```
int height;
// while printf is intended for output
// scanf is intended for accepting inputs
scanf(“%d”, &height);
```

the address of operator

Reading User Input

Character

```
int character;
// getchar() is an appropriate choice for
// reading single character
character = getchar();
// putchar() displays the character
putchar(character);
```

Reading User Input

String

```
// This declaration says that string is
// a character variable that can hold
// 10 characters!

char string[10];
```

Reading User Input

String

```
// This declaration says that string is
// a character variable that can hold
// 10 characters!

char string[10];
```

Look at the square brackets - Remember this, it will keep coming where you want to use a collection of data

Reading User Input

String

```
// This declaration says that string is
// a character variable that can hold
// 10 characters!

char string[10];
gets(string); // get the string
puts(string); // put the string
```

15CSE102
Computer Programming

(Next Topic)

$$A + B = C$$