

Variables, tipos, constantes

Variables

- Sirven para almacenar:
 - Valores de entrada
 - Valores de salida
 - Valores intermedios
 - Valores modificados

Sintaxis de variables en C++

- Es necesario declarar una variable antes de utilizarla
- Los nombres de las variables sólo pueden contener caracteres alfanuméricos (0..9, a..z, A..Z) y el carácter *underscore* (`_`)
- El primer carácter del nombre no puede ser numérico

Ejemplos de variables en C++

- Legales:
 - x, y, xyz_Hello, bob, joe12, ____a1____, CamelCase
- No legales:
 - 12joe, hi-there, go!

Consejos

- Declarar las variables al principio de la función donde van a ser utilizadas (excepto variables de bucle for)
- Utilizar nombres con significado
 - Bien: age, tirePressure, lateral_distortion
 - Mal: x
- Utilizar comentarios para dar más información
 - `int age; // how long the tire has been installed on the car`

Tipos números enteros: `short`

- Ejemplos de declaraciones:
 - `short shoe_size = 6;`
 - `unsigned short age;`
- Tamaño en memoria:
 - 2 bytes
- Rango de valores posibles:
 - Con signo: -32768 .. 32767
 - Sin signo: 0 .. 65535

Tipos números enteros: `int/long`

- Ejemplos de declaraciones:
 - `int numCarsOnHwy;`
 - `unsigned long numBoatsOnLake = 5001U;`
- Tamaño en memoria:
 - 4 bytes
- Rango de valores posibles:
 - Con signo: -2147483648 .. 2147483647
 - Sin signo: 0 .. 4294967295

Tipos números enteros: long long

- Ejemplos de declaraciones:
 - `long long desk_length = 123LL;`
 - `unsigned long long numKeys;`
- Tamaño en memoria:
 - 8 bytes
- Rango de valores posibles:
 - Con signo: $-2^{63} .. 2^{63} - 1$
 - Sin signo: $0 .. 2^{64} - 1$

Tipos números reales: float

- Ejemplos de declaraciones:

- `float bodyTemp = 98.6;`
- `float money_in_pocket;`

- Tamaño en memoria:

- 4 bytes

- Precisión:

- 6 dígitos significativos

Tipos números reales: double

- Ejemplos de declaraciones:
 - `double weight_of_mountain = 746538433.55; // tons`
 - `double max_speed;`
- Tamaño en memoria:
 - 8 bytes
- Precisión:
 - 15 dígitos significativos

Tipos números reales: **long double**

- Ejemplos de declaraciones:
 - `long double feather_wt = 0.000000000000000032; // lbs`
 - `long double planck_const;`
- Tamaño en memoria:
 - 12 bytes (no estandarizado)
- Precisión:
 - Depende del tamaño, en cualquier caso mucho mayor que la del tipo `double`




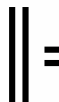












Tipos no numéricos: char

- Ejemplos de declaraciones:
 - `char continueResponse = 'y';`
 - `unsigned char one_byte;`
- Tamaño en memoria:
 - 1 byte
- Rango de valores posibles:
 - De 0 a 255, correspondientes al set de caracteres ASCII

Set de caracteres ASCII básico

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	Null	32	20	Space	64	40	@	96	60	`
1	01	Start of heading	33	21	!	65	41	A	97	61	a
2	02	Start of text	34	22	"	66	42	B	98	62	b
3	03	End of text	35	23	#	67	43	C	99	63	c
4	04	End of transmit	36	24	\$	68	44	D	100	64	d
5	05	Enquiry	37	25	%	69	45	E	101	65	e
6	06	Acknowledge	38	26	&	70	46	F	102	66	f
7	07	Audible bell	39	27	'	71	47	G	103	67	g
8	08	Backspace	40	28	(72	48	H	104	68	h
9	09	Horizontal tab	41	29)	73	49	I	105	69	i
10	0A	Line feed	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage return	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	47	2F	/	79	4F	O	111	6F	o
16	10	Data link escape	48	30	0	80	50	P	112	70	p
17	11	Device control 1	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	50	32	2	82	52	R	114	72	r
19	13	Device control 3	51	33	3	83	53	S	115	73	s
20	14	Device control 4	52	34	4	84	54	T	116	74	t
21	15	Neg. acknowledge	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	54	36	6	86	56	V	118	76	v
23	17	End trans. block	55	37	7	87	57	W	119	77	w
24	18	Cancel	56	38	8	88	58	X	120	78	x
25	19	End of medium	57	39	9	89	59	Y	121	79	y
26	1A	Substitution	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	59	3B	;	91	5B	[123	7B	{
28	1C	File separator	60	3C	<	92	5C	\	124	7C	
29	1D	Group separator	61	3D	=	93	5D]	125	7D	}
30	1E	Record separator	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	63	3F	?	95	5F	_	127	7F	□

Set de caracteres ASCII extendido

ÇüéâäàåçêëèìîïÄÅÉæÆôöòûùÿÖÜø£Ø×ƒáíó
úñÑªº¿®¬½¼¡«»|÷ÁÂÀ©¢¥
ãÃ \mathbb{L}  \propto øÐÊËÈıÍÎÏ \lrcorner óßÔÒõÕμƒƆ
ÚÛÜÝÝ[—]´±₌^¾¶§÷‚°¨·¹³² \blacksquare

Tipos no numéricos: `string`

- Ejemplos de declaraciones:
 - `string firstName = "Clayton";`
- Tamaño en memoria:
 - 1 byte por cada carácter, más 1 byte de finalización
- Utilizar siempre comillas dobles para `string` y comillas simples para `char`

Tipos no numéricos: `bool`

- Ejemplos de declaraciones:
 - `bool quit = false;`
 - `bool passed = true;`
- Tamaño en memoria:
 - 1 byte
- Las palabras `true` y `false` son palabras reservadas en C++

Tipos enumerados

// sintaxis

```
enum <enum_name> {enumerator1, enumerator2, ... enumeratorN};
```

// ejemplos

```
enum color {GREEN, BLUE, YELLOW};
```

```
enum number {ONE = 1, TWO, THREE, FOUR, FIVE};
```

```
enum exit {DOOR = 7, WINDOW = 3, CHIMNEY, DRAIN = 1};
```

Ejemplo

```
enum color {GREEN, BLUE, YELLOW};
```

```
enum color hue1;
```

```
enum color hue2 = GREEN;
```

```
enum color hue3 = 2;
```

Constantes

- Debemos usarlas cuando el valor no puede/debe ser alterado
- Misma sintaxis que las variables, precedida de la palabra reservada **const**
- Ejemplos:
 - `const float PI = 3.14159;`
 - `const double DISTORTION_COEFFICIENT = 5.66239;`
 - `const unsigned int DAYS_PER_WEEK = 7;`
 - `PI = 4; // Así no, error de compilación`

Consejos para nombrar constantes

```
const short TWO = 2;                // mala elección
float val1, val2, average;
average = (val1 + val2) / TWO;
```

```
const short TWO = 3;                // peor incluso
float val1, val2, val3, average;
average = (val1 + val2 + val3) / TWO;
```

```
const short DIVISOR = 2;            // mejor, más explícito
float val1, val2, average;
average = (val1 + val2) / DIVISOR;
```

Palabras reservadas

- Básicamente palabras clave e identificadores predefinidos
- No debemos darles ningún otro uso
- Ejemplos de lo que no debemos hacer:
 - `int new;`
 - `float main;`
 - `string cout;`
 - `double case = 0.1;`

Palabras reservadas

```
auto      const      double     float      int       short     struct
unsigned  break      continue   else       for       long
signed    switch     void      case      default   enum      goto
register   sizeof     typedef   volatile   char      do
extern     if         return    static    union     while     asm
dynamic_cast  namespace  reinterpret_cast  bool
explicit   new        static_cast  typeid    catch     false
try        operator   template    typename  class     friend
private    this         using       const_cast  inline    public
throw      virtual    delete     mutable    protected true
wchar_t    and        bitand      compl      not_eq    or_eq
xor_eq     and_eq     bitor       not        or        xor      cin
endl       INT_MIN    iomanip     main       npos     std      cout
include    INT_MAX   ostream     MAX_RAND   NULL
string
```