

E/S estàndard en C++ :Introducció

- Flux d'entrada: serie d'entrades que alimenten els programes (algoritmes) en un ordinador.
- Flux d'eixida: serie d'exides que el programa genera. ...
Com anem a mostrar el resultats?
- Per a nosaltres:
 - Entrada estandar: teclat.
 - Eixida estandar: pantalla.

Eixides amb cout

- Podem enviar a la pantalla qualsevol combinació de variables y cadenes:

```
cout << num_dulces << "dulces\n"
```

```
cout << num_dulces;  
cout << "dulces\n"
```

```
cout << num_dulces << " dulces" << endl;
```

... incloure expresions aritmetiques:

```
cout << "El precio total es:" << (precio1 + precio2)
```

Operador d' inserció

Eixides amb cout

- Secuencias amb caràcters especials: totes comencen amb el caràcter '\'.
- Algunes secuencias ...

\n	Nova línia
\t	Tabulació horitzontal
\\	Diagonal invertida
\"	Cometes dobles

- Exemple:

```
//enviar una llinea en blanc a la pantalla
cout << "\n";
cout << endl;
```

Funcions de flux per al format d'eixida

- **Format:** l'organització de l'eixida dels nostres programes.

- *Mètodes* associats al flux d'eixida (cout.<metode>):

- *precision*: Fixa el nombre de cifres darrere la coma.

```
cout.precision(2);
```

- *width*: indica al flux quants espais ha d'usar al enviar un element a l'eixida.

Només afecten a la següent
eixida per pantalla

```
cout.width(4);
```

- *fill*: indica el caràcter amb el qual es completarà els espais en blanc al enviar un element a l'eixida.

```
cout.fill('a');
```

... mètodes del format d'eixida (banderes)

- `setf`: (set flags) estableix banderes. El flux es vorà afectat en funció de la/es bandera/es actives.

<code>ios::fixed</code>	Formato de punto fijo
<code>ios::showpoint</code>	Incluya un punto decimal en números en coma flotante.
<code>ios::showpos</code>	Signo + para números positivos
<code>ios::left</code>	Siguiente número en el extremo izquierdo.
<code>ios::right</code>	Siguiente número en el extremo derecho.

**Les banderes
permaneixen actives
tot el Programa?
Sí, encara q es poden
desactivar ...**

.. format d'eixida (banderes)

- `unsetf`: quan se vol bandera activa es pot desactivar amb aquesta funció membre.
- Exemples:

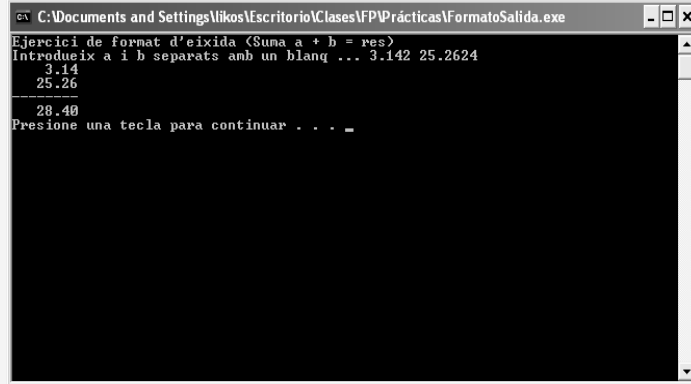
```
cout.setf(ios::fixed);
cout.setf(ios::showpos);
cout.setf(ios::showpoint);
cout.setf(ios::left);
cout.setf(ios::right);
```

Activa

```
cout.unsetf(ios::showpos);
```

Desactiva

Ejercici :



```

C:\Documents and Settings\likos\Escritorio\Clases\FP\Prácticas\FormatoSalida.exe
Ejercici de format d'eixida (Suma a + b = res)
Introdueix a i b separats amb un blanc ... 3.142 25.2624
  3.14
 25.26
-----
28.40
Presione una tecla para continuar . . . =

```

Ejercici :

```

#include <iostream>
#include <stdlib.h>

int main(int argc, char *argv[])
{
    float a, b, res;

    cout << "Ejercici de format d'eixida (Suma a + b = res) " << endl;
    cout << "Introdueix a i b separats amb un blanc ... " ;
    cin >> a >> b;

    res = a + b;

    cout.precision(2);
    cout.setf(ios::fixed);
    cout.setf(ios::right);
    // cout.setf(ios::showpoint); ... per defecte

    cout.width(8); cout.fill(' '); cout << a << endl;
    cout.width(8); cout.fill(' '); cout << b << endl;
    cout << "-----" << endl;
    cout.width(8); cout.fill(' '); cout << res << endl;

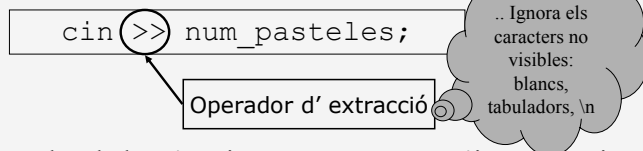
    system("pause");

    return 0;
}

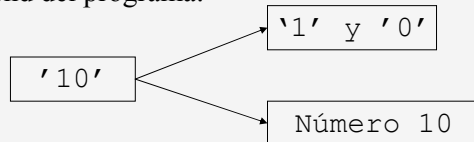
```

Entrades amb cin

- `cin` es el flux d'entrada estàndar (teclat). Exemple:



- Totes les dades s'envien com caràcters. L'interpretació depend del programa:

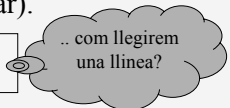


Entrades i eixides de caràcters

- .. més mètodes de `cin` i `cout` :

- `get`: permet llegir un caràcter d'entrada i guardarlo dins d'una variable de tipo caràcter (`char`).

```
char siguiente_simbolo;
cin.get(siguiente_simbolo);
```



- `put`: envia un caràcter a l'eixida.

```
char siguiente_simbolo='a';
cout.put(siguiente_simbolo);
```

- `ignore`: ignora num caràcters mentrestant no trobe el caràcter `delim`

```
cin.ignore(int num, int delim);
cin.ignore(); // ignora un caràcter
```

Entrades i eixides de caràcters

- Comparació entre `cin >> i` i `cin.get`:

cin.get

```
char c1,c2,c3,c4;

//Introduce 4 caracteres separados por espacios
cin.get(c1);
cin.get(c2);
cin.get(c3);
cin.get(c4);

//Imprimo los caracteres leídos
cout << "Los cuatro caracteres leídos son:<< endl;
cout.put(c1);
cout.put(c2);
cout.put(c3);
cout.put(c4);
```

Entrades i eixides de caràcters

- Comparació entre `cin >> i` i `cin.get`:

cin >>

```
char c1,c2,c3,c4;

//Prueba con cin
//Introduce 4 caracteres separados por espacios
cin >> c1;
cin >> c2;
cin >> c1;
cin >> c2;

//Imprimo los caracteres leídos
cout << "Los cuatro caracteres leídos son:<< endl;
cout << c1 << c2 << c3 << c4;
```

```
C:\E:\clases\FundaProg\curso02-03\ejemplos\entrada.exe
```

```
Introduce cuatro caracteres separados por espacios
1 2 3 4
Los cuatro caracteres leídos son:
1 2 Presione una tecla para continuar . . . _
```

```
C:\E:\clases\FundaProg\curso02-03\ejemplos\entrada2.exe
```

```
Introduce 4 caracteres separados por espacios
1 2 3 4
Los cuatro caracteres leídos son:
1234
Presione una tecla para continuar . . . _
```

Exercici final:

- Escriure un programa en C++ que ens demane dues dates (dd mm aa) i ens calcule la seua diferència en segons.

```
C:\Documents and Settings\Administrator\Desktop\fechas.exe
```

```
Introducir fecha inicial <dd mm aa>: 1 1 01
```

```
Introducir fecha final <dd mm aa>: 16 10 02
```

Día	Mes	Año
1	1	1
16	10	2
15 días	09 meses	01 años

```
Transcurridos 56937600 segundos
```

```
... pulsa para salir
```

Exercici(II): justificar
les columnes a la dreta

<pre> #include <iostream.h> #include <stdio.h> #include <stdlib.h> int main(int argc, char *argv[]) { int dd_i, mm_i, aa_i; int dd_f, mm_f, aa_f; int num_segundos; cout << "Introducir fecha inicial (dd mm aa): "; cin >> dd_i >> mm_i >> aa_i; cout << endl << "Introducir fecha final (dd mm aa): "; cin >> dd_f >> mm_f >> aa_f; //Calculo del total en segundos num_segundos = ((aa_f - aa_i)*365*24*60*60) + ((mm_f - mm_i)*31*24*60*60) + ((dd_f - dd_i)*24*60*60); // Salida ... cout << endl; cout << "tDias:t'tMeses:t'tAnyos" << endl; cout << "t-----t't-----" << endl; </pre>	<pre> cout << "t" << dd_i << "t" << mm_i << "t" << aa_i << endl; cout << "t" << dd_f << "t" << mm_f << "t" << aa_f << endl; cout << "t-----t't-----" << endl; cout << "t"; cout.width(2); cout.fill('0'); cout << dd_f - dd_i << " dias "; cout << "t"; cout.width(2); cout.fill('0'); cout << mm_f - mm_i << " meses "; cout << "t"; cout.width(2); cout.fill('0'); cout << aa_f - aa_i << " anyos" << endl; cout << endl << "t Transcursos " << num_segundos << " segundos" << endl; cout << endl << endl << "t t... pulsa para salir" << endl; system("PAUSE"); return 0; } </pre>
---	---

Podem sumar dos characters?

```
C:\Documents and Settings\Nikos\Escritorio\Clases\
Dona'm 2 caracters: + +
+
+
+
U
Presione una tecla para continuar . . . _
```

```
#include <iostream>

int main(int argc, char *argv[])
{
    char a, b, res;

    cout << "Dona'm 2 characters: ";
    cin >> a >> b;

    res = a + b;

    cout << a << endl;
    cout << b << endl;
    cout << "-----" << endl;
    cout << res << endl;
    system("pause");
}
```

```
return 0;
}
```

ASCII	Hex	Simbolo	ASCII	Hex	Simbolo	ASCII	Hex	Simbolo	ASCII	Hex	Simbolo
0	0	NUL	16	10	DLE	32	20	(espacio)	48	30	0
1	1	SOH	17	11	DC1	33	21	!	49	31	1
2	2	STX	18	12	DC2	34	22	"	50	32	2
3	3	ETX	19	13	DC3	35	23	#	51	33	3
4	4	EOF	20	14	DC4	36	24	\$	52	34	4
5	5	END	21	15	NAK	37	25	%	53	35	5
6	6	ACK	22	16	SYN	38	26	&	54	36	6
7	7	BEL	23	17	ETB	39	27	'	55	37	7
8	8	BS	24	18	CAN	40	28	(	56	38	8
9	9	TAB	25	19	EM	41	29	)	57	39	9
10	A	LF	26	1A	SUB	42	2A	*	58	3A	*
11	B	VT	27	1B	ESC	43	2B	+	59	3B	+
12	C	FF	28	1C	FS	44	2C	,	60	3C	,
13	D	OR	29	1D	GS	45	2D	-	61	3D	-
14	E	SO	30	1E	RS	46	2E	.	62	3E	.
15	F	SI	31	1F	US	47	2F	/	63	3F	/

ASCII	Hex	Simbolo	ASCII	Hex	Simbolo	ASCII	Hex	Simbolo	ASCII	Hex	Simbolo
64	40	@	80	50	P	96	60	'	112	70	p
65	41	A	81	51	Q	97	61	a	113	71	q
66	42	B	82	52	R	98	62	b	114	72	r
67	43	C	83	53	S	99	63	c	115	73	s
68	44	D	84	54	T	100	64	d	116	74	t
69	45	E	85	55	U	101	65	e	117	75	u
70	46	F	86	56	V	102	66	f	118	76	v
71	47	G	87	57	W	103	67	g	119	77	w
72	48	H	88	58	X	104	68	h	120	78	x
73	49	I	89	59	Y	105	69	i	121	79	y
74	4A	J	90	5A	Z	106	6A	j	122	7A	z
75	4B	K	91	5B	[	107	6B	k	123	7B	{
76	4C	L	92	5C	\	108	6C	l	124	7C	
77	4D	M	93	5D	]	109	6D	m	125	7D	}
78	4E	N	94	5E	^	110	6E	n	126	7E	~
79	4F	O	95	5F	_	111	6F	o	127	7F	?

Tabla de códigos ASCII - Formato de caracteres estándares