

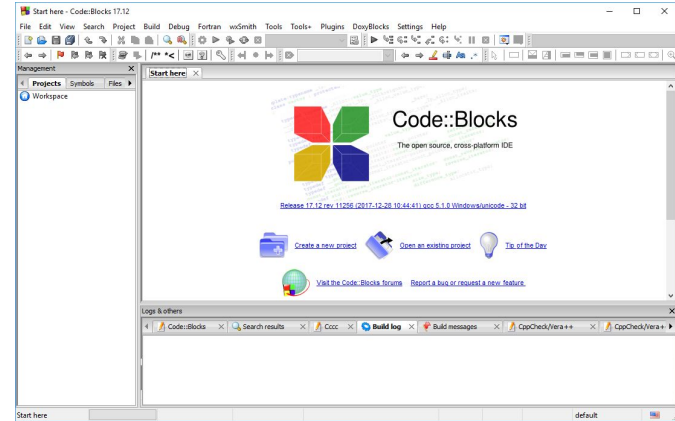
Programmation C

1 - Introduction

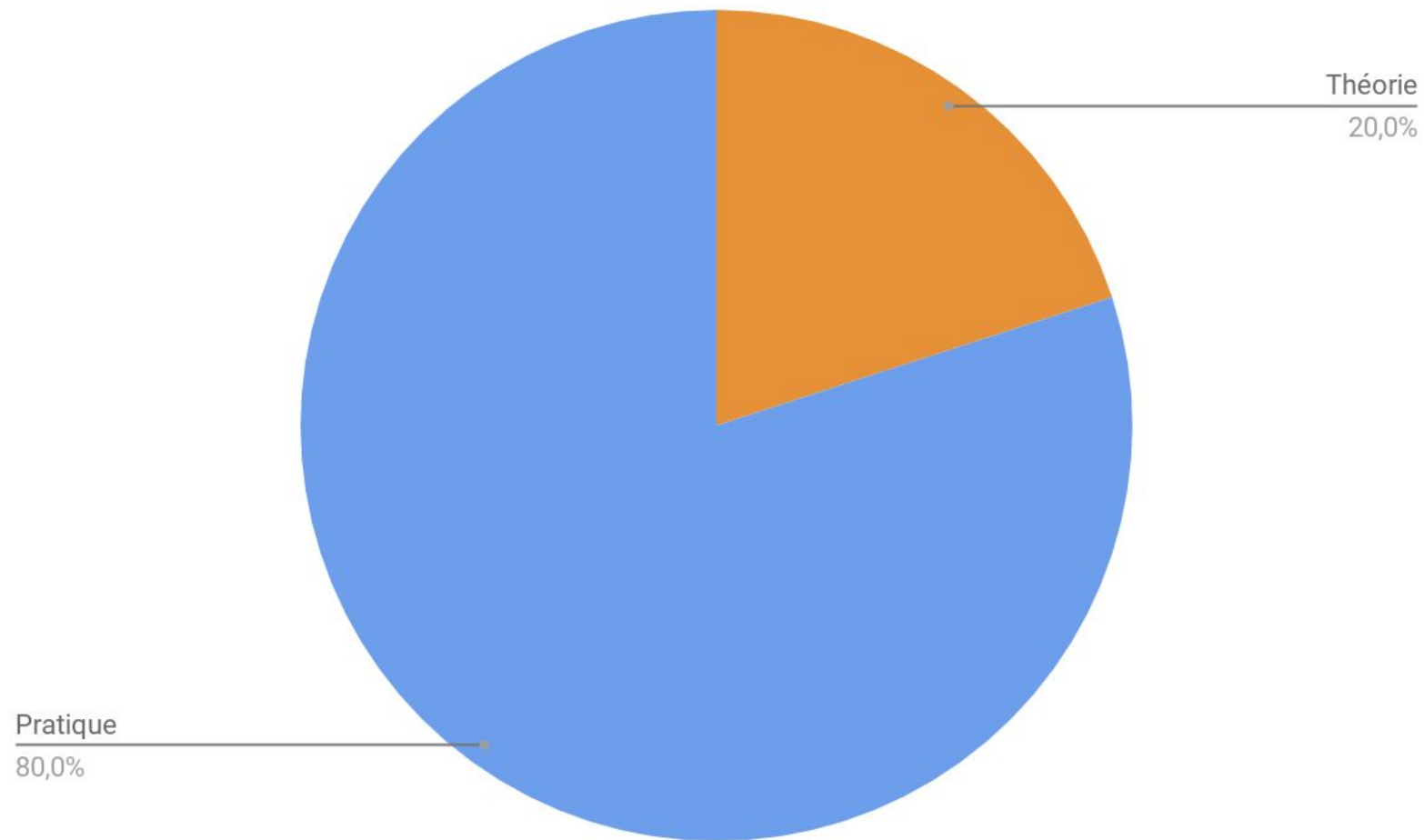
Programme et Logiciel



Jeux vidéos



Editeur de code



Ecrire du code c'est du gâteau



Ingrédients:

- 4 oeufs.
- 200g de sucre.
- 200g de beurre.
- 200g de farine.

Instructions:

1. Allumer le four pour le préchauffer à 150°C.
2. Mélanger les ingrédients dans un saladier.
3. Verser le contenu du saladier dans un moule à gâteau.
4. Quand le four est chaud, enfourner le gâteau.
5. Attendre 40 minutes de cuisson.
6. Sortir le gâteau du four.
7. Laisser refroidir le gâteau.
8. Démouler le gâteau.
9. Miam !

Du code au programme

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int vie = 3;
    int game_over = 0;

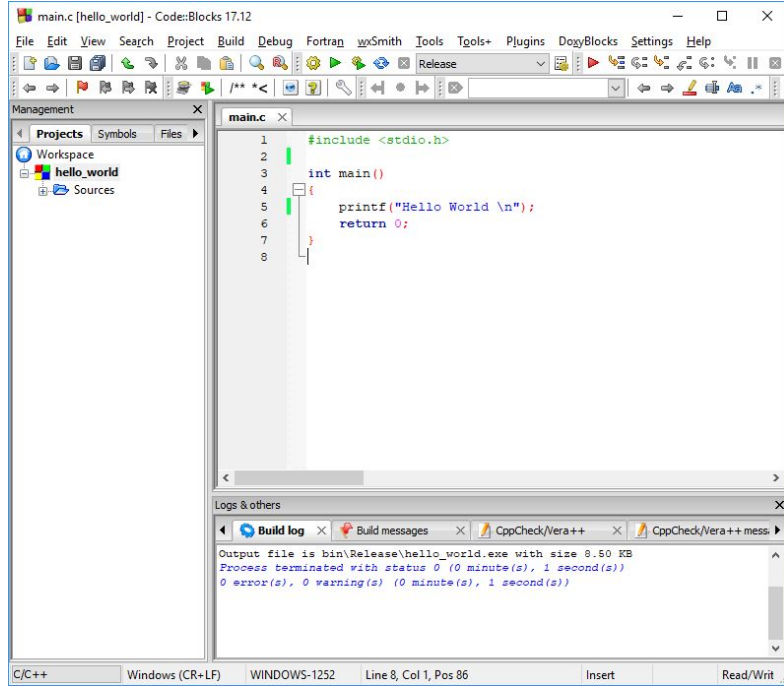
    InitialiserLeJeu();
    AfficherMenuDemarage();
    ...
}
```

Code source



Programme exécutable

Hello World



The screenshot shows the Code::Blocks IDE interface. The main editor window displays the source code for a C program named 'main.c'. The code includes the standard input/output header and a main function that prints 'Hello World' and returns 0. The left sidebar shows the project structure with 'hello_world' as the active project. The bottom status bar indicates the current line and column.

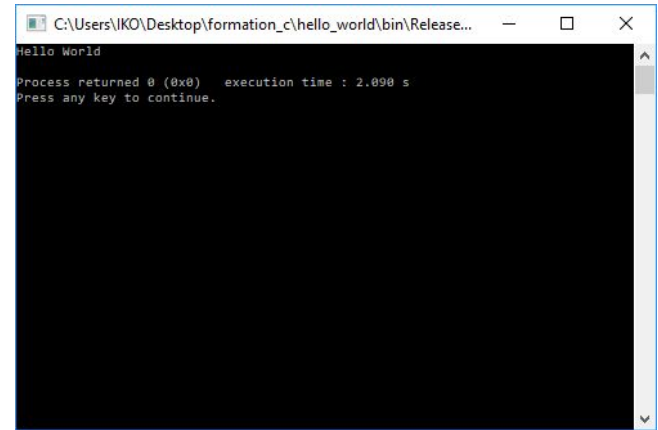
```
1 #include <stdio.h>
2
3 int main()
4 {
5     printf("Hello World \n");
6     return 0;
7 }
8
```

Build log output:

```
Output file is bin\Release\hello_world.exe with size 8.60 KB
Process terminated with status 0 (0 minute(s), 1 second(s))
0 error(s), 0 warning(s) (0 minute(s), 1 second(s))
```

Code source

compilateur



The screenshot shows a command prompt window titled 'C:\Users\IKO\Desktop\formation_c\hello_world\bin\Release...'. It displays the output of the compiled program, which is 'Hello World'. Below the output, it shows the process return code (0) and execution time (2.090 s).

```
Hello World
Process returned 0 (0x0)   execution time : 2.090 s
Press any key to continue.
```

Programme exécutable

Le compilateur

**Langage Humainement
compréhensible**

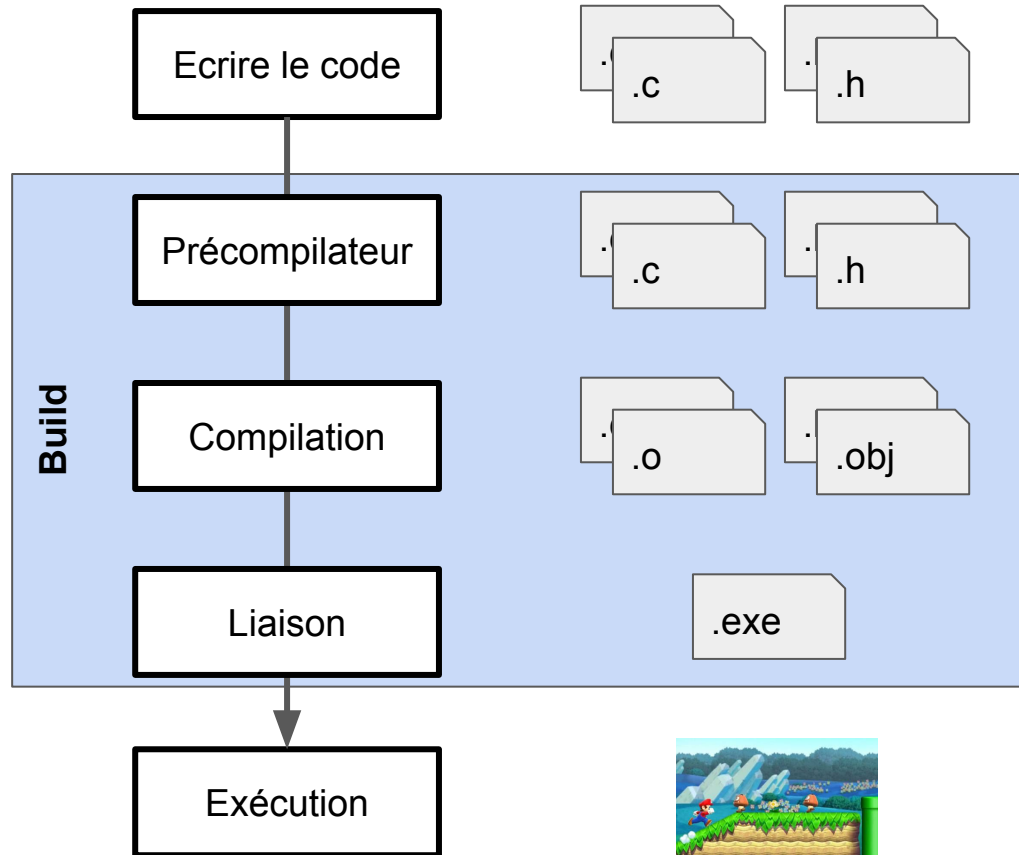
exemple:
Afficher "Hello" à l'écran

Compilateur

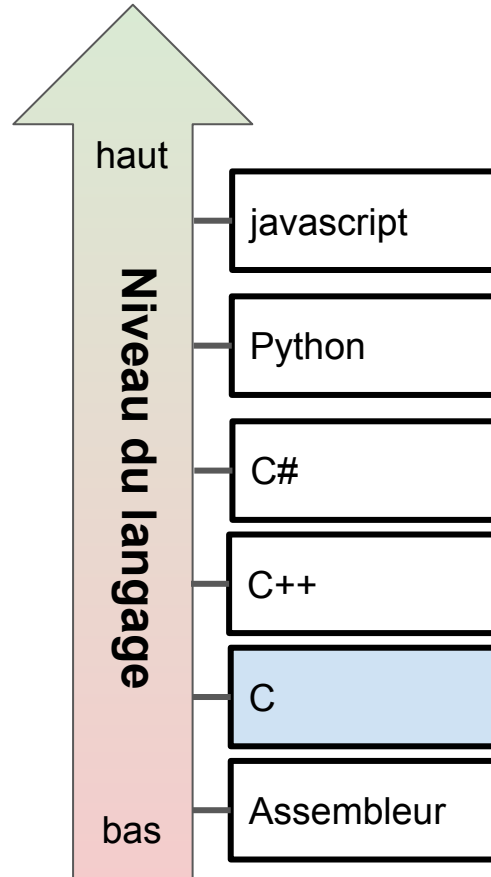
**Langage binaire compris
par les ordinateurs**

exemple:
011110011101101011100011

Création d'un Logiciel



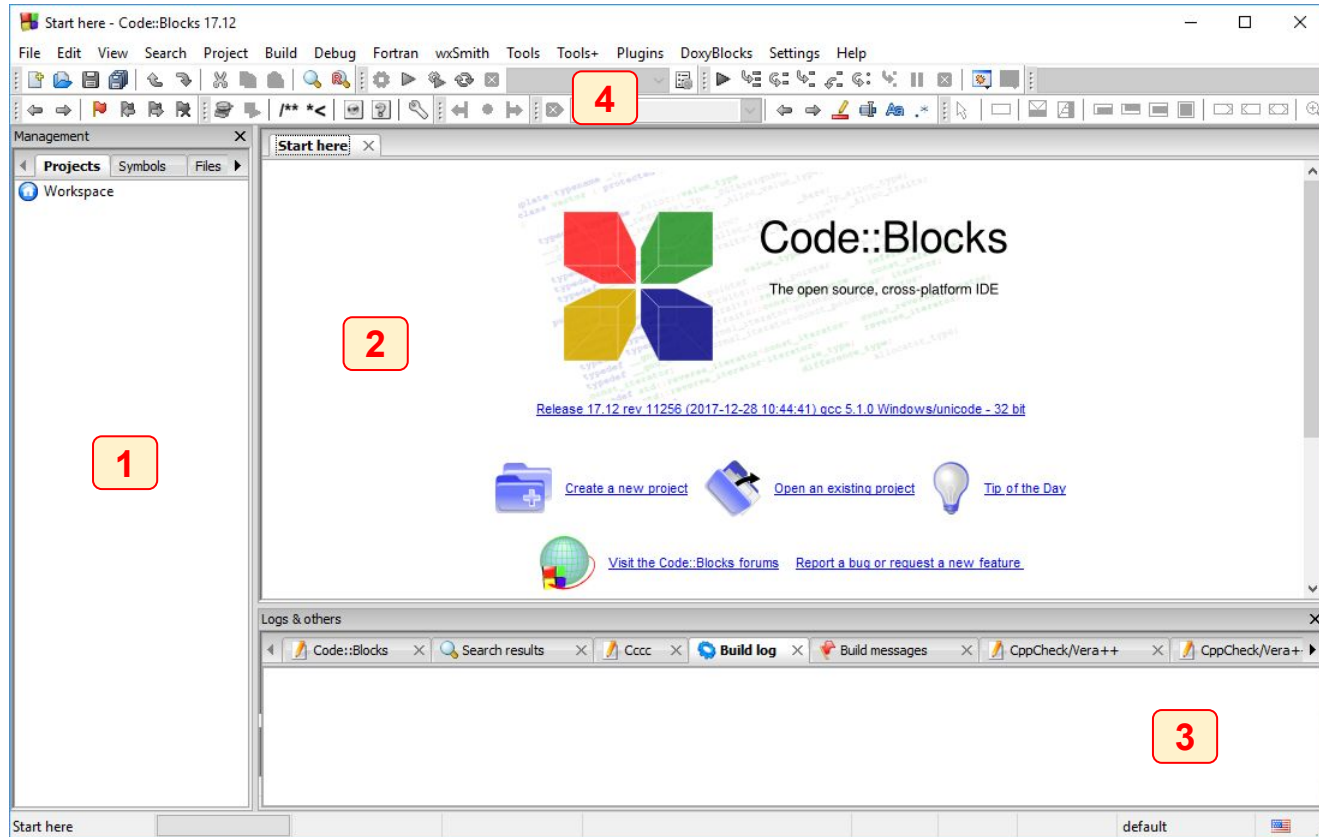
Niveau des langages



Programmation C

2 - Mise en place

code::blocks



Téléchargement: <http://www.codeblocks.org/downloads/binaries>

Recompiler le code

Modifier le code

Build

main.c

```
mingw32-gcc.exe -Wall -O2 -c main.c -o obj\Release\main.o
```

main.o

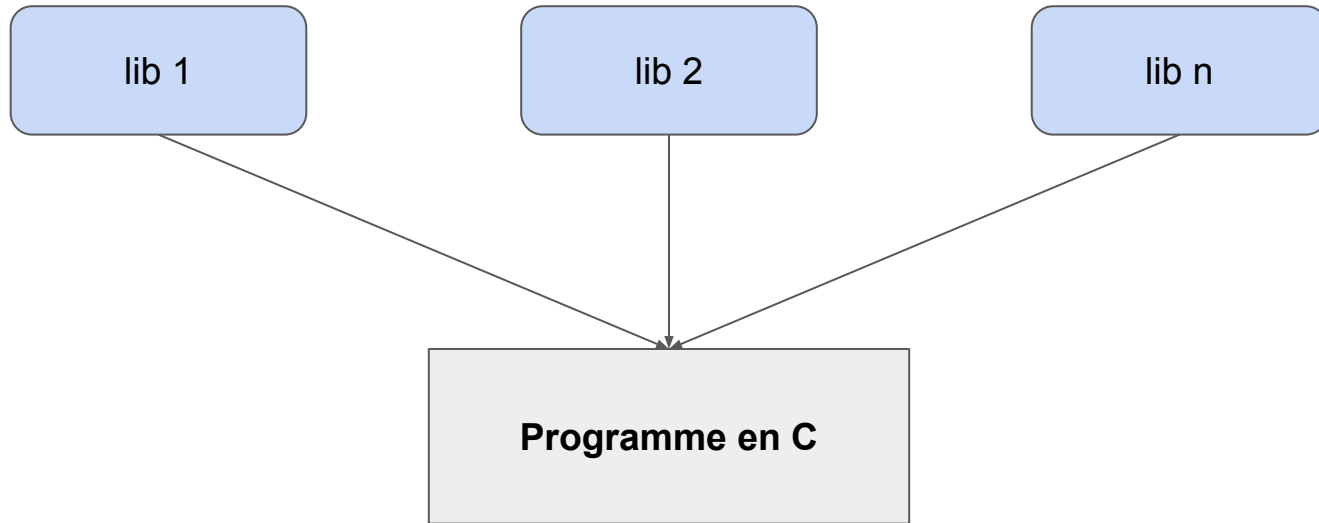
```
mingw32-g++.exe -o bin\Release\hello_world.exe obj\Release\main.o -s
```

main.exe

Programmation C

3 - Syntaxe et Structure

Le C et les libs



Contenue de la STL

STL (Standard Template Library)

stdio.h

stdlib.h

math.h

...

Documentation de la STL: <https://devdocs.io/c>

Commentaires

// commentaire sur une ligne

```
/*  
  bloc de  
  commentaires  
*/
```

```
// Affiche un message à l'écran  
printf("Hello world! \n");
```

Précompilateur

```
printf("Hello world! \n");
```


Programmation C

4 - Mémoire et Variables

Exécuter un programme

Disque dure

mon_programme.exe



Chargement

RAM (mémoire vive)

mon programme

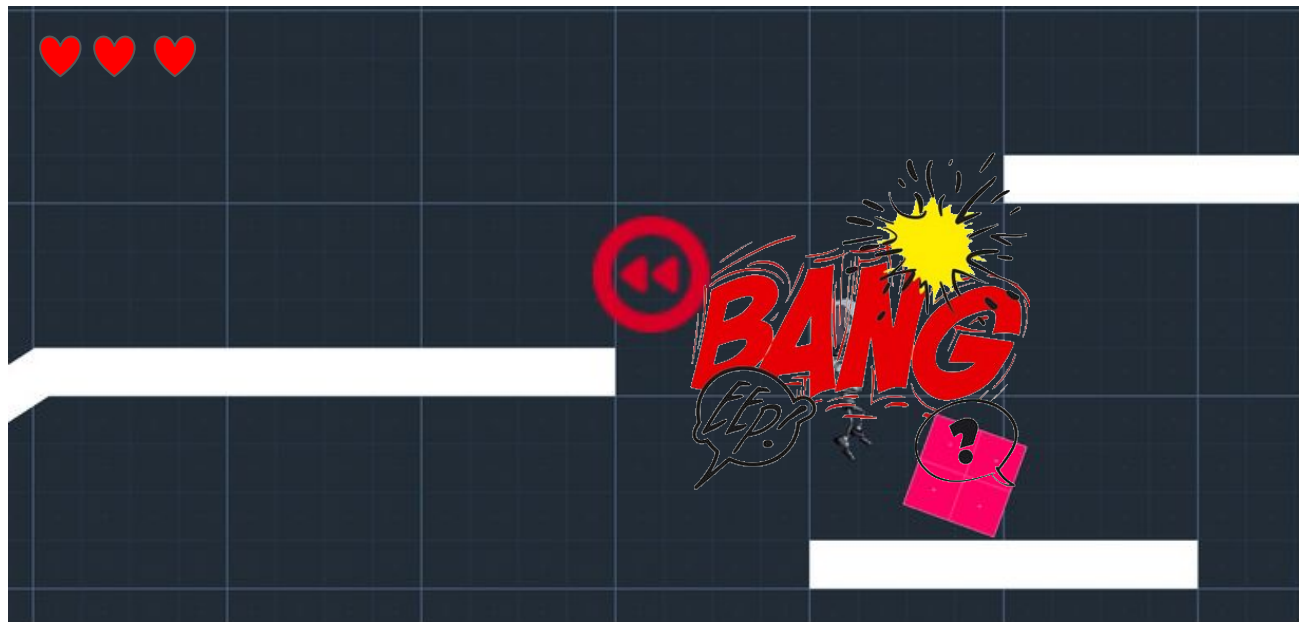
autre programme



Utilisation de la RAM

MÉMOIRE

vie = 2



Gestion de la mémoire

Adresse: 10 11 12 13 14 15 16 17 18 19

MÉMOIRE

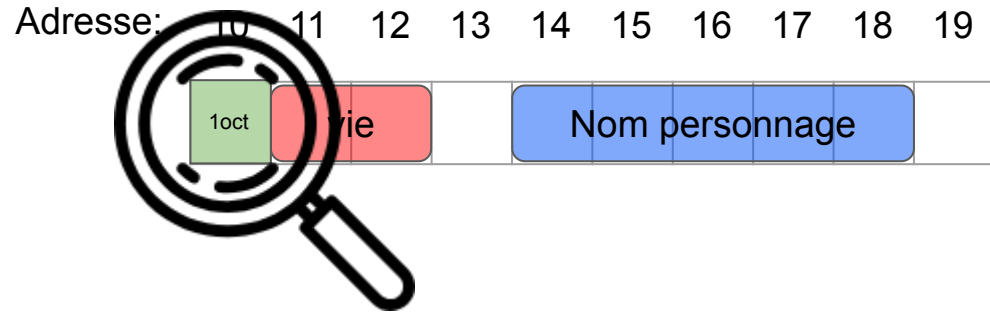


	vie		Nom personnage	
--	-----	--	----------------	--



Gestion de la mémoire

MÉMOIRE



1oct

=

8 bits

ex: 1001 0111

Déclaration de variable

type

nom_de_variable

=

valeur

;

Exemple:

```
int ma_variable = 0;
```

```
int vie = 3;
```

ram



Règles de nommage des variables

Règles

Pas de nombre en début de nom

Pas de caractères spéciaux à l'exception de _

Pas de mots réservés en c (ex type de variable)

Pas d'accent (à, é, è , ...)

Pas d'espace

Nom explicite (très important pour la lisibilité)

tout en minuscule avec _ pour séparer les mots

Exemples

6variable

nom-de-variable

int

début

nom de variable

NomDeVariable

nom_de_variable

Mots réservés en C

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

Les types de base

Type	Taille (octets)	Valeur min	Valeur max	Affichage
int	2 / 4	-32768 / -2147483648	32767 / 2147483647	%d
char	1	-128	127	%c
float	4	-3.4e38	3.4e38	%f
double	8	-1.7e308	1.7e308	%f

Les types signed et unsigned

Type	Taille (octets)	Valeur min	Valeur max	Affichage
unsigned int	2 / 4	0 / 0	65535/ 4294967295	%d
signed int	2 / 4	-32768 / -2147483648	32767 / 2147483647	%d
int	2 / 4	-32768 / -2147483648	32767 / 2147483647	%d
unsigned char	1	0	255	%c
signed char	1	-128	127	%c
char	1	-128	127	%c
float	4	-3.4e38	3.4e38	%f
double	8	-1.7e308	1.7e308	%f

Les types short et long

Type	Taille (octets)	Valeur min	Valeur max	Affichage
unsigned short int / unsigned short	2	0	65535	%d
short int / short	2	-32768	32767	%d
unsigned long int / unsigned long	4	0	4294967295	%ld
long int / long	4	-2147483648	2147483647	%ld
unsigned int	2 / 4	0 / 0	65535/ 4294967295	%d
int	2 / 4	-32768 / -2147483648	32767 / 2147483647	%d
unsigned char	1	0	255	%c
signed char	1	-128	127	%c
char	1	-128 / 0	127 / 255	%c
float	4	-3.4e38	3.4e38	%f
double	8	-1.7e308	1.7e308	%f

Programmation C

5 - Opérateurs et variables

Les opérateurs

Symbole	Signification	Exemple
=	Assignment	ma_variable = 12
+	Addition	ma_variable + 3
-	Soustraction	ma_variable - 1
*	Multiplication	ma_variable * 2
/	Division	ma_variable / 100
%	Modulo	ma_variable % 2

$$5 \% 2 = 1 \quad \Rightarrow \quad 2 * 2 + 1$$

Les opérateurs spéciaux

Symbole	Exemple	Équivalent
+=	ma_variable += 5	ma_variable = ma_variable + 5
-=	ma_variable -= 5	ma_variable = ma_variable - 5
*=	ma_variable *= 5	ma_variable = ma_variable * 5
/=	ma_variable /= 5	ma_variable = ma_variable / 5
%=	ma_variable %= 5	ma_variable = ma_variable % 5

Les incréments et décréments

Symbole	Exemple	Équivalent
++	ma_variable ++	ma_variable = ma_variable + 1
--	ma_variable --	ma_variable = ma_variable - 1

Programmation C

6 - Complément sur les variables

Les constantes

const type NOM_DE_CONSTANTE = valeur ;

Exemple:

```
const int TVA = 20;  
const int MAX_VIES = 3;
```



Règles de nommage identique au variable mais tout en majuscule

Table ASCII

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	\$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[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	[NEGATIVE 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	[ENG OF 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	[SUBSTITUTE]	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	[DEL]

Overflows

	Valeur min (dec)	Valeur min (bin)	Valeur max (dec)	Valeur max (dec)
unsigned short (2 octets)	0	0000 0000 0000 0000	65535	1111 1111 1111 1111



```
unsigned short ma_variable = 0;  
ma_variable --;  
  
printf("ma_variable = %d", ma_variable);
```

Exemple

ma_variable = 65535