

Mục đích của lập trình C

Xây dựng các nền tảng tính toán.



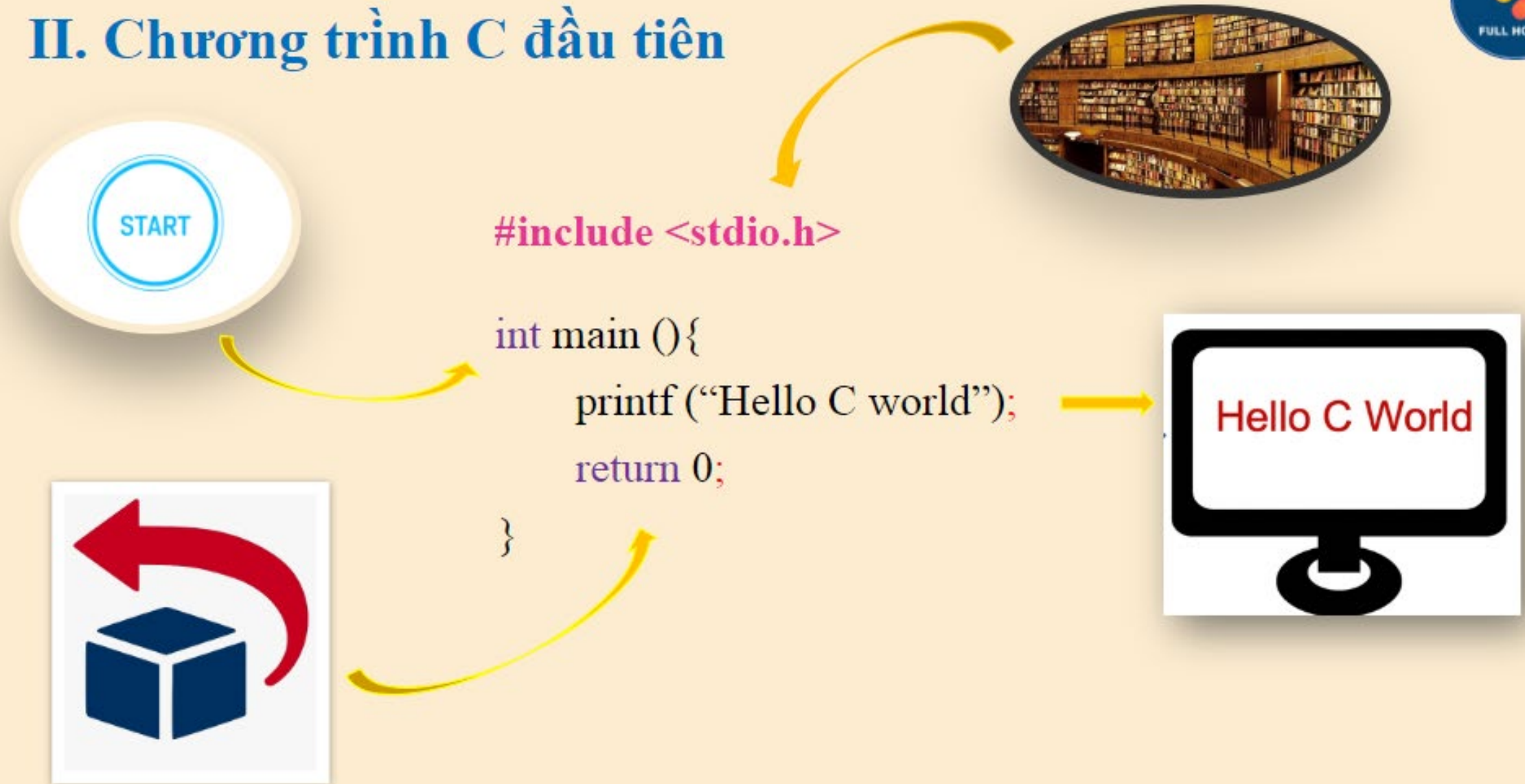
Thiết kế phần mềm hệ thống.



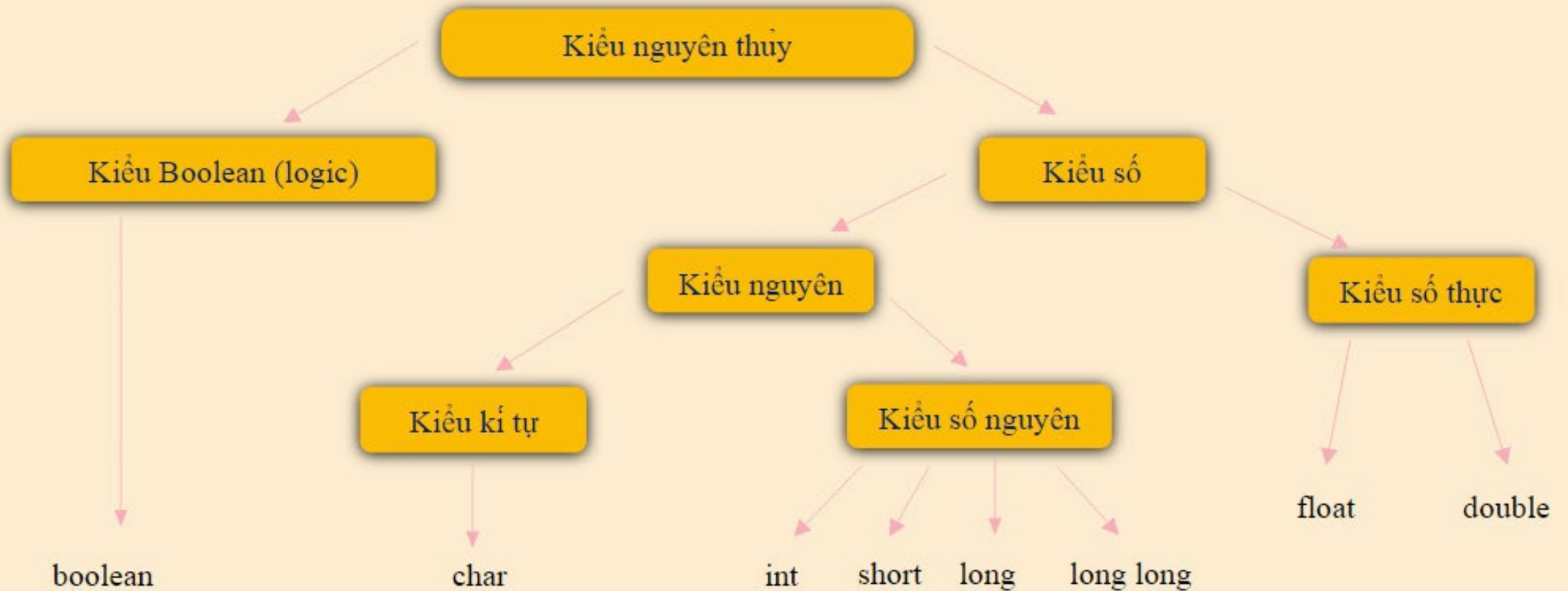
Phát triển ngôn ngữ mới



II. Chương trình C đầu tiên



III. Kiểu dữ liệu trong C



1, Kiểu số nguyên

Kiểu	Kích thước	Vùng giá trị
char	1 byte	-128 tới 127 hoặc 0 tới 255
unsigned char	1 byte	0 tới 255
signed char	1 byte	-128 tới 127
int	2 hoặc 4 bytes	-32,768 tới 32,767 hoặc -2,147,483,648 tới 2,147,483,647
unsigned int	2 hoặc 4 bytes	0 tới 65,535 hoặc 0 tới 4,294,967,295
short	2 bytes	-32,768 tới 32,767
unsigned short	2 bytes	0 tới 65,535
long	4 bytes	-2,147,483,648 tới 2,147,483,647
unsigned long	4 bytes	0 tới 4,294,967,295
long long	8 bytes	-9223372036854775808 tới 9223372036854775807
unsigned long long	8 bytes	0 tới 18446744073709551615

1, Kiểu số nguyên

2, Kiểu số thực

Type	Kích thước	Phạm vi giá trị	Độ chính xác
float	4 byte	$1.2\text{E}-38$ to $3.4\text{E}+38$	6 chữ số thập phân
double	8 byte	$2.3\text{E}-308$ to $1.7\text{E}+308$	15 chữ số thập phân
long double	10 byte	$3.4\text{E}-4932$ to $1.1\text{E}+4932$	19 chữ số thập phân

1, Kiểu số nguyên

2, Kiểu số thực

3, Kiểu void



3. Các kiểu dữ liệu hay dùng và đặc tả của nó.

%d: số nguyên hệ 10 có dấu

%u: số nguyên hệ 10 không dấu

%x: số nguyên hệ 16

%o: số nguyên hệ bát phân

%s: xâu kí tự

%c: một kí tự đơn

%f: số chấm động cố định

%e: số chấm động (ký hiệu có số mũ)

l : Tiền tố dùng kèm với %d, %x, %o để chỉ số nguyên dài (ví dụ %ld)

Khái niệm

Biến trong C

Là tên của 1 vị trí bộ nhớ

Dùng để lưu trữ giá trị dữ liệu

Có thể thay đổi và dùng lại
nhiều lần

Biến trong C

Khái niệm

Cú pháp khai báo
biến

`kiểu_dữ_liệu danh_sách_biến;`

Biến trong C

Khái niệm

Cú pháp khai báo biến

Quy tắc khai báo biến trong C

Một biến có thể có các chữ cái, chữ số và dấu gạch dưới.

Tên biến chỉ có thể bắt đầu bằng chữ cái hoặc dấu gạch dưới.

Không có khoảng trắng trong tên biến.

Tên biến không phải là từ khóa dành riêng như int, float,.....

Tên biến có phân biệt hoa thường.

Biến trong C

Khái niệm

Cú pháp khai báo
biến

Quy tắc khai báo
biến trong C

```
int a, b, c;  
float _hb;  
char xin6;
```

Biến trong C

Khái niệm

Cú pháp khai báo
biến

Quy tắc khai báo
biến trong C

Các kiểu biến
thường gặp trong C

Biến trong C

Khái niệm

Cú pháp khai báo biến

Quy tắc khai báo biến trong C

Các kiểu biến thường gặp trong C

```
3 int main(){  
4     int A;  
5     char c;  
6     return 0;  
7 }
```

Biến local
(địa phương).

Biến trong C

Khái niệm

Cú pháp khai báo biến

Quy tắc khai báo biến trong C

Các kiểu biến thường gặp trong C

```
3 int heluu;  
4 float k22;  
5  
6 int main(){  
7  
8     return 0;  
9 }
```

Biến global
(toàn cầu).



V. Cách lấy kiểu dữ liệu và kích thước

\n : Xuống dòng

\t : Tab ngang

\r : Nhảy về đầu hàng

\a : Kêu Bíp

**** : In ra dấu \

\" : In ra dấu “

\' : In ra dấu ‘

%% : In ra dấu %

VI. Một số toán tử cơ bản trong C

1. Một số phép toán cơ bản

Toán tử	Miêu tả
+	Cộng
-	Trừ
*	Nhân
/	Chia
%	Lấy dư
<	Nhỏ hơn
<=	Nhỏ hơn hoặc bằng
>	Lớn hơn
>=	Lớn hơn hoặc bằng

VI. Một số toán tử cơ bản trong C

1. Một số phép toán cơ bản

2. Toán tử tăng giảm

Toán tử	Miêu tả	Ví dụ	
		Tiền tố	Hậu tố
++	Tăng lên 1 đơn vị	++a;	a++;
		Tương đương $a = a + 1$;	
--	Trừ đi 1 đơn vị	--a;	a--;
		Tương đương $a = a - 1$;	

VI. Một số toán tử cơ bản trong C

1. Một số phép toán cơ bản

2. Toán tử tăng giảm

3. Toán tử gán

Toán tử	Miêu tả	Ví dụ
=	Gán giá trị toán hạng bên phải cho toán hạng trái .	$a = 10;$ $a = b + c;$
+=	Thêm giá trị toán hạng phải tới toán hạng trái và gán giá trị đó cho toán hạng trái.	$a += c;$ tương đương $a = a + c;$
-=	Trừ đi giá trị toán hạng phải từ toán hạng trái và gán giá trị này cho toán hạng trái.	$a -= c;$ tương đương $a = a - c;$
*=	Nhân giá trị toán hạng phải với toán hạng trái và gán giá trị này cho toán hạng trái.	$a *= c;$ tương đương $a = a * c;$
/=	Chia toán hạng trái cho toán hạng phải và gán giá trị này cho toán hạng trái.	$a /= c;$ tương đương $a = a / c;$
%=	Lấy phần dư của phép chia toán hạng trái cho toán hạng phải và gán cho toán hạng trái.	$a \% = c;$ tương đương $a = a \% c;$

VI. Một số toán tử cơ bản trong C

1. Một số phép toán cơ bản
2. Toán tử tăng giảm
3. Toán tử gán
4. Các phép toán logic

Toán tử	Miêu tả
&&	Và
 	Hoặc
!=	Khác

VI. Một số toán tử cơ bản trong C

1. Một số phép toán cơ bản
2. Toán tử tăng giảm
3. Toán tử gán
4. Các phép toán logic
5. Toán tử điều kiện

`biểu_thức_1 ? biểu_thức_2 : biểu_thức_3;`



VI. Một số toán tử cơ bản trong C

1. Một số phép toán cơ bản
2. Toán tử tăng giảm
3. Toán tử gán
4. Các phép toán logic
5. Toán tử điều kiện
6. Các ngôi trong toán tử

Một ngôi	Hai ngôi	Ba ngôi
++, --	+, -, *, /, %, !=, &&,	? :

VII. Một số chương trình cơ bản

```
2 //ket hop dac ta
3 #include <stdio.h>
4
5 int main() {
6     int a = 15, b=10;
7     printf("a/b = %d",a/b);
8     printf("\na/b = %f",a/b);
9     printf("\na/b = %.2f",(float)a/b);
10    return 0;
11 }
```

VII. Một số chương trình cơ bản

```
3 //ham scanf
4 #include <stdio.h>
5
6 int main() {
7     int a;
8     float b;
9     char c;
10    scanf("%c%d%f",&c,&a,&b);
11    printf("%c %d %.2f",c,a,b);
12    return 0;
13 }
```

Ký tự và bảng mã ASCII

Bảng mã
ASCII
là gì?

Ký tự và bảng mã ASCII

Bảng mã
ASCII
là gì?

Đây là bộ mã hóa ký tự
cho
bảng chữ cái La Tinh và
được dùng để hiển thị v
ấn bản trong máy tính.

Giúp máy tính có hiểu
và hiển thị được các ký
t
ự mà bạn muốn nhập
vào máy tính .



ASCII Table

Dec	Hex	Oct	Char	Dec	Hex	Oct	Char	Dec	Hex	Oct	Char	Dec	Hex	Oct	Char
0	0	0		32	20	40	[space]	64	40	100	@	96	60	140	`
1	1	1		33	21	41	!	65	41	101	A	97	61	141	a
2	2	2		34	22	42	"	66	42	102	B	98	62	142	b
3	3	3		35	23	43	#	67	43	103	C	99	63	143	c
4	4	4		36	24	44	\$	68	44	104	D	100	64	144	d
5	5	5		37	25	45	%	69	45	105	E	101	65	145	e
6	6	6		38	26	46	&	70	46	106	F	102	66	146	f
7	7	7		39	27	47	'	71	47	107	G	103	67	147	g
8	8	10		40	28	50	(72	48	110	H	104	68	150	h
9	9	11		41	29	51)	73	49	111	I	105	69	151	i
10	A	12		42	2A	52	*	74	4A	112	J	106	6A	152	j
11	B	13		43	2B	53	+	75	4B	113	K	107	6B	153	k
12	C	14		44	2C	54	,	76	4C	114	L	108	6C	154	l
13	D	15		45	2D	55	-	77	4D	115	M	109	6D	155	m
14	E	16		46	2E	56	.	78	4E	116	N	110	6E	156	n
15	F	17		47	2F	57	/	79	4F	117	O	111	6F	157	o
16	10	20		48	30	60	0	80	50	120	P	112	70	160	p
17	11	21		49	31	61	1	81	51	121	Q	113	71	161	q
18	12	22		50	32	62	2	82	52	122	R	114	72	162	r
19	13	23		51	33	63	3	83	53	123	S	115	73	163	s
20	14	24		52	34	64	4	84	54	124	T	116	74	164	t
21	15	25		53	35	65	5	85	55	125	U	117	75	165	u
22	16	26		54	36	66	6	86	56	126	V	118	76	166	v
23	17	27		55	37	67	7	87	57	127	W	119	77	167	w
24	18	30		56	38	70	8	88	58	130	X	120	78	170	x
25	19	31		57	39	71	9	89	59	131	Y	121	79	171	y
26	1A	32		58	3A	72	:	90	5A	132	Z	122	7A	172	z
27	1B	33		59	3B	73	;	91	5B	133	[123	7B	173	{
28	1C	34		60	3C	74	<	92	5C	134	\	124	7C	174	
29	1D	35		61	3D	75	=	93	5D	135]	125	7D	175	}
30	1E	36		62	3E	76	>	94	5E	136	^	126	7E	176	~
31	1F	37		63	3F	77	?	95	5F	137	_	127	7F	177	

VIII. Ký tự và bảng mã ASCII

1. Bảng mã ASCII là gì?

2. Ví dụ.

```
3  #include <stdio.h>
4
5  int main(){
6      char c;
7      scanf("%c",&c);
8      printf("%c %d",c,c);
9      return 0;
10 }
```

IX. Câu lệnh điều khiển trong C

1. Câu lệnh **if**.

a. Cấu trúc

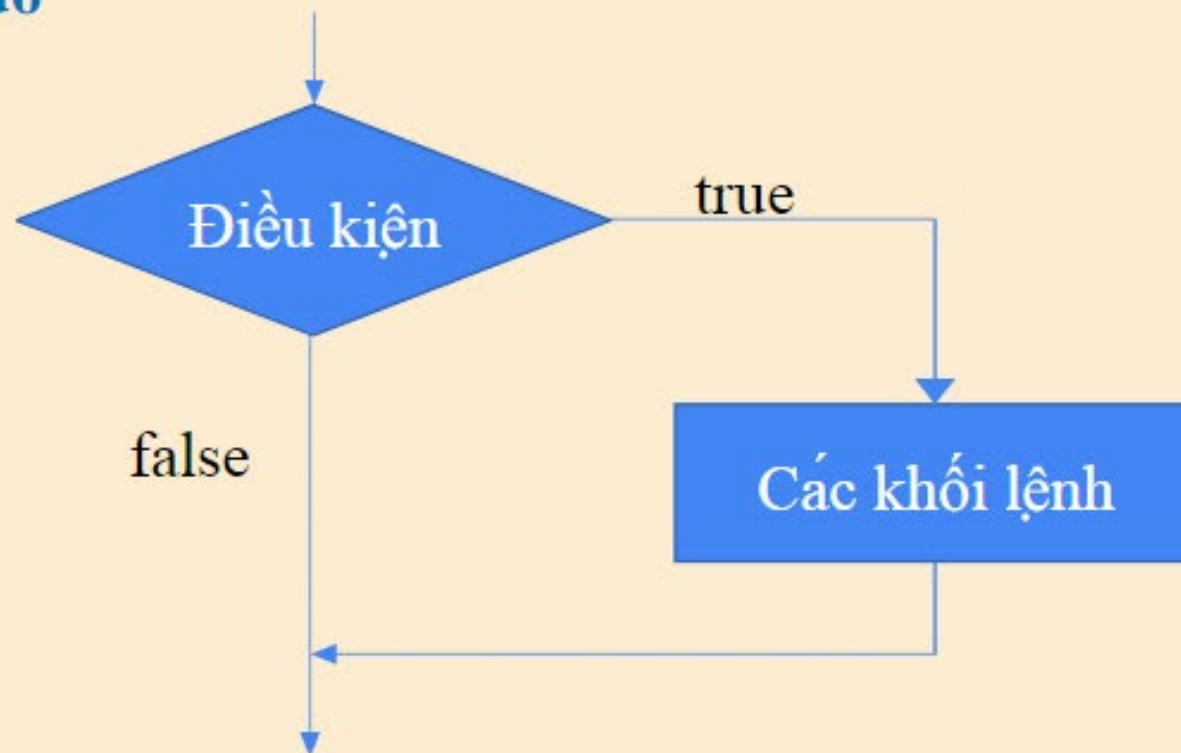
```
if (điều kiện){  
    // Khối lệnh sẽ được thực hiện nếu <điều kiện> đúng.  
}
```

IX. Câu lệnh điều khiển trong C

1. Câu lệnh **if**.

a. Cấu trúc

b. Lưu đồ



IX. Câu lệnh điều khiển trong C

1. Câu lệnh **if**.

a. Cấu trúc

b. Lưu đồ

c. Ví dụ

```
1  #include <stdio.h>
2
3  int main() {
4      int a;
5      printf("Nhap a = ");
6      scanf("%d", &a);
7
8      if (a % 2 != 0) {
9          printf("%d la so le", a);
10     }
11     printf("\nHoan thanh!");
12 }
```

IX. Câu lệnh điều khiển trong C

1. Câu lệnh **if else**.

a. Cấu trúc

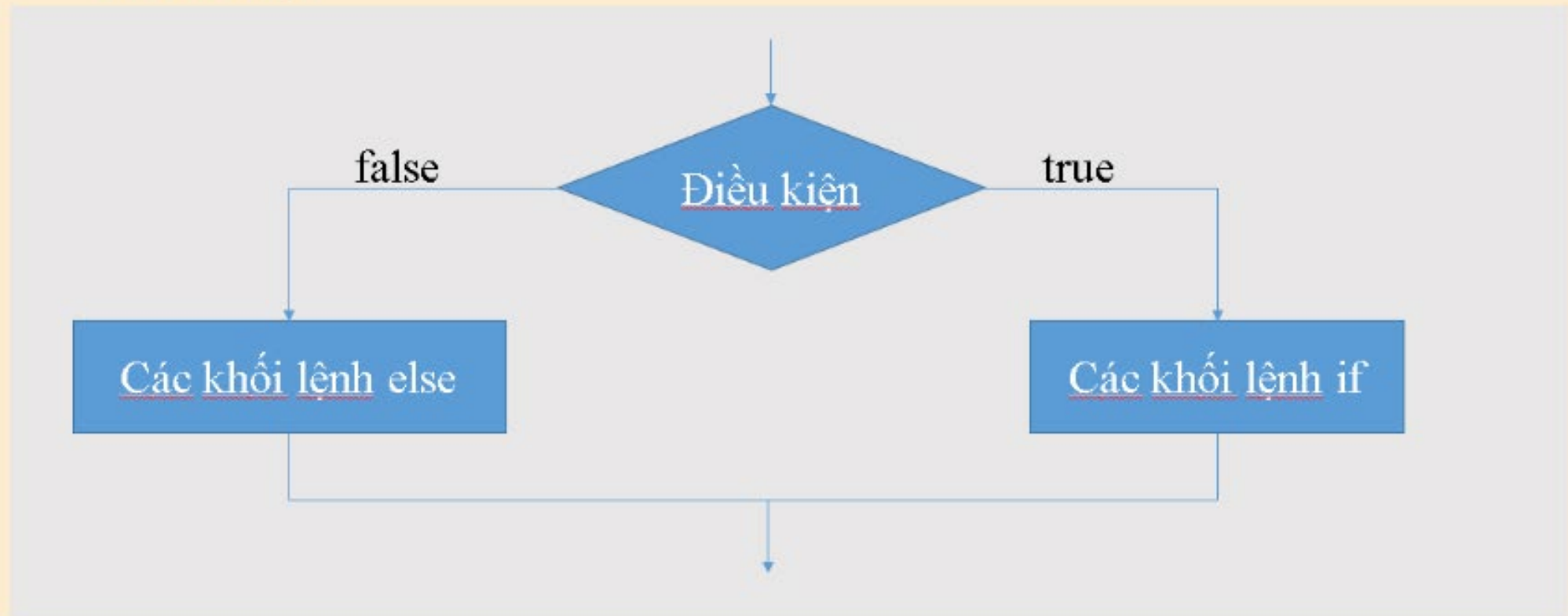
```
if (điều kiện){  
    // Khối lệnh sẽ được thực hiện nếu <điều kiện> đúng.  
} else {  
    // Khối lệnh sẽ được thực hiện nếu <điều kiện> sai.  
}
```

IX. Câu lệnh điều khiển trong C

1. Câu lệnh **if else**.

a. Cấu trúc

b. Lưu đồ



IX. Câu lệnh điều khiển trong C

1. Câu lệnh **if else**.

a. Cấu trúc

b. Lưu đồ

c. Ví dụ

```
3 #include <stdio.h>
4
5 int main(){
6     int n;
7     scanf("%d",&n);
8     if(n%5==0){
9         printf("%d chia het cho 5",n);
10    }else {
11        printf("%d khong chia het cho 5",n);
12    }
13    return 0;
14 }
```

IX. Câu lệnh điều khiển trong C

1. Câu lệnh **If else if else.**

a. Cấu trúc

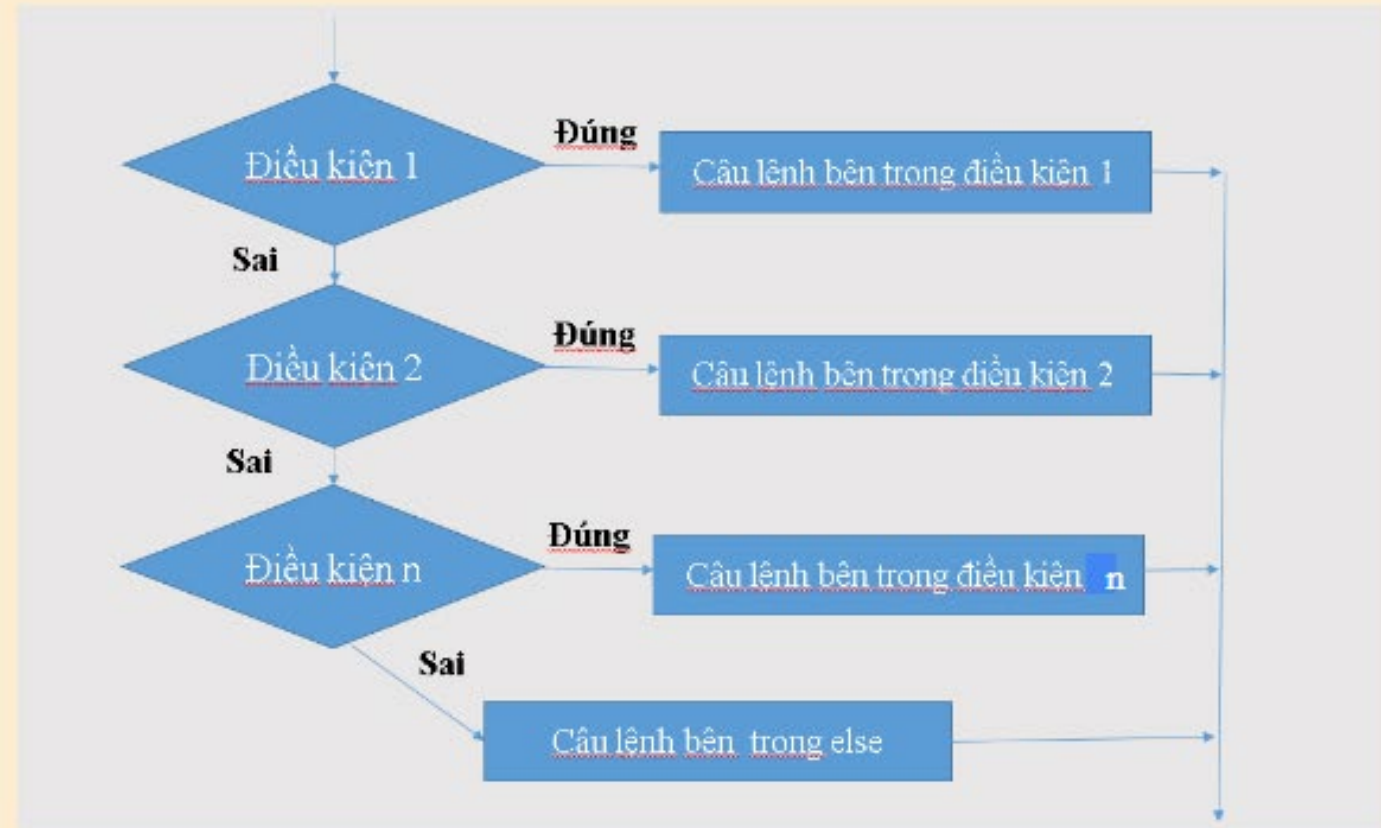
```
if (điều kiện 1){  
    // Khối lệnh sẽ được thực hiện nếu <điều kiện1> đúng.  
}else if (điều kiện 2){  
    // Khối lệnh sẽ được thực hiện nếu <điều kiện 2> đúng.  
}else if (điều kiện 3){  
    // Khối lệnh sẽ được thực hiện nếu <điều kiện 3> đúng.  
}  
.  
.  
else{  
    // Khối lệnh sẽ được thực hiện nếu tất cả điều kiện sai.  
}
```

IX. Câu lệnh điều khiển trong C

1. Câu lệnh **If else if else.**

a. Cấu trúc

b. Lưu đồ



IX. Câu lệnh điều khiển trong C

1. Câu lệnh **If else if else.**

a. Cấu trúc

b. Lưu đồ

c. Ví dụ

```
1  #include <stdio.h>
2
3  int main() {
4      int a,b;
5      scanf("%d%d", &a, &b);
6
7      if (a > b) {
8          printf("%d > %d", a,b);
9      }else if (a < b){
10         printf("%d < %d", a,b);
11     }else {
12         printf("%d = %d", a,b);
13     }
14 }
```

IX. Câu lệnh điều khiển trong C

1. Câu lệnh **switch case**.

a. Cấu trúc

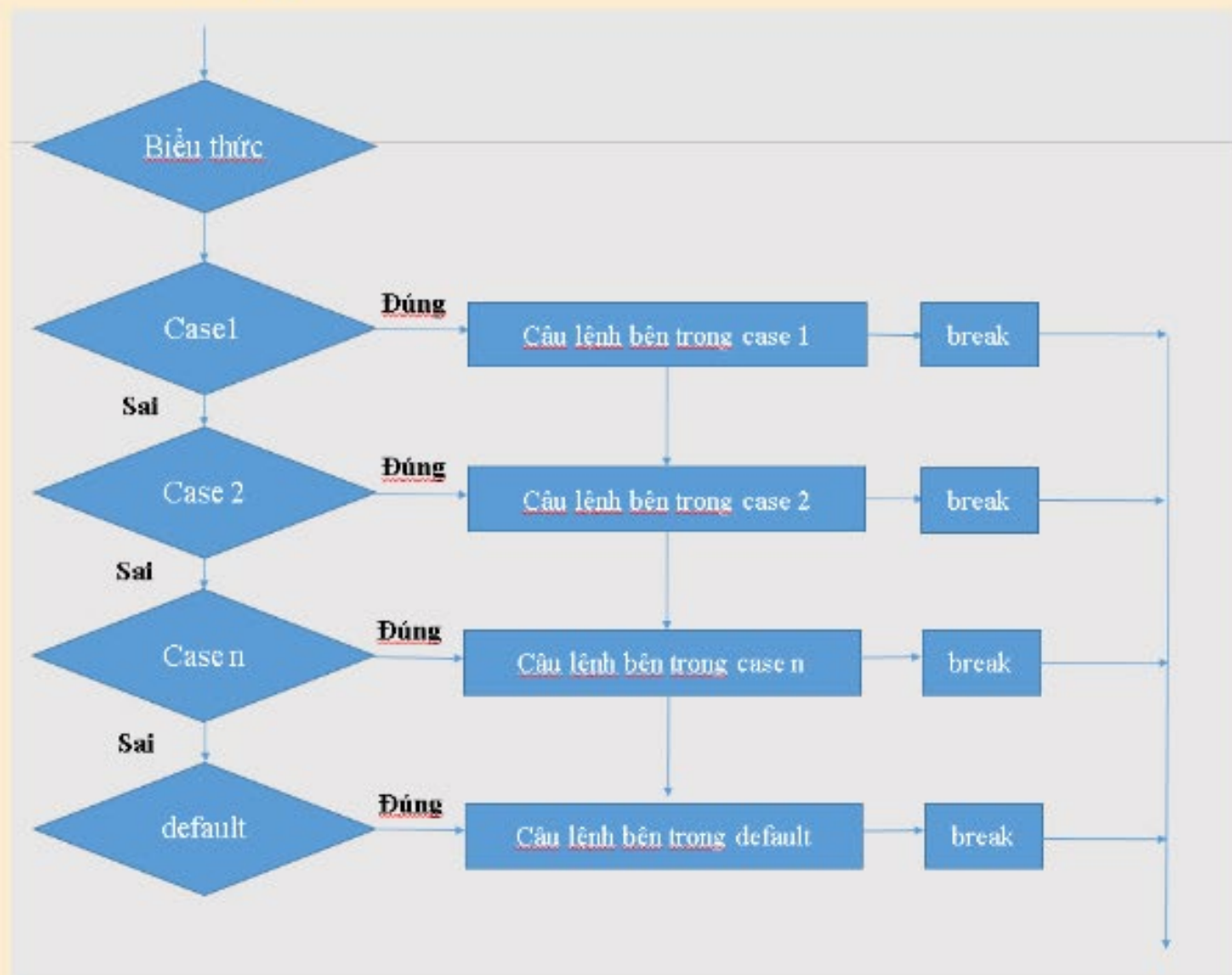
```
switch (bieu_thuc) {  
    case gia_tri_1:  
        // Khởi lệnh 1  
        break; //tùy chọn  
    case gia_tri_2:  
        // Khởi lệnh 2  
        break; //tùy chọn  
    .....  
    case gia_tri_n:  
        // Khởi lệnh n  
        break; //tùy chọn  
    default:  
        // Khởi lệnh  
}
```

IX. Câu lệnh điều khiển trong C

1. Câu lệnh **switch case**.

a. Cấu trúc

b. Lưu đồ



IX. Câu lệnh điều khiển trong C

1. Câu lệnh **switch case**.

a. Cấu trúc

b. Lưu đồ

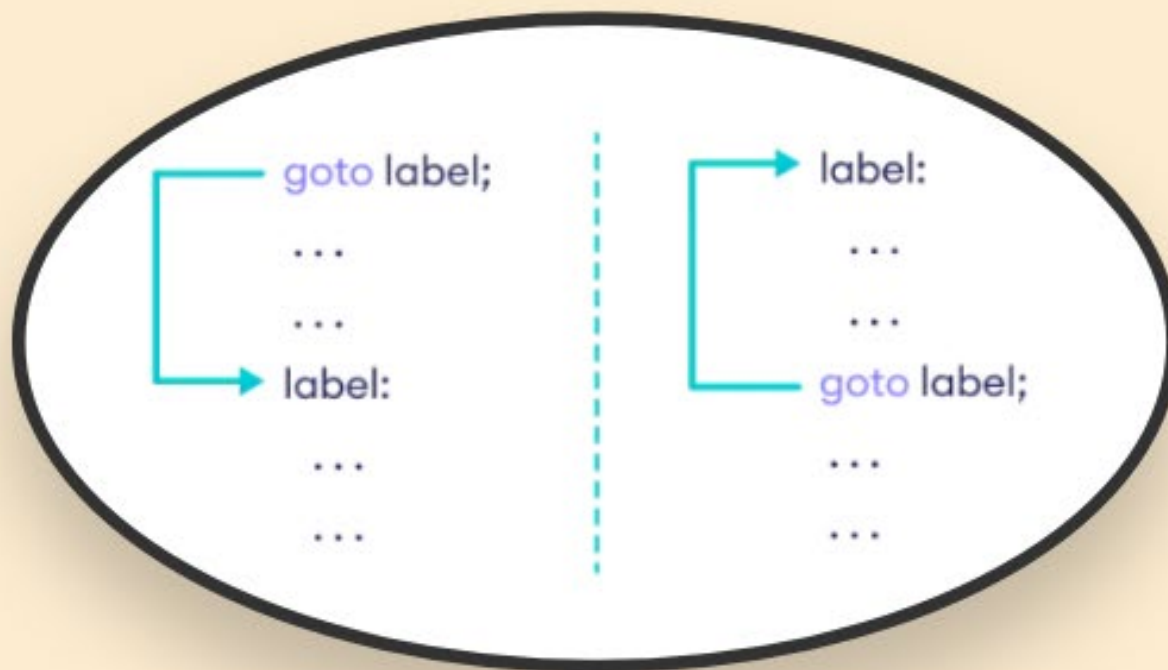
c. Ví dụ

```
1  #include <stdio.h>
2
3  int main() {
4      int a;
5      scanf("%d", &a);
6
7      switch (a){
8          case 10:
9              printf("10");
10         case 11:
11             printf("\n11");
12         case 12:
13             printf("\n12");
14             break;
15         case 13:
16             printf("\n13");
17         default:
18             printf("\nKhac!");
19     }
20 }
21
```


IX. Câu lệnh điều khiển trong C

1. Câu lệnh **goto**.

a. Cấu trúc



IX. Câu lệnh điều khiển trong C

1. Câu lệnh **goto**.

a. Cấu trúc

b. Ví dụ

```
1  #include <stdio.h>
2
3  int main(){
4      int n;
5      nhap:
6          printf("Xin chao moi nguoi\n");
7          scanf("%d",&n);
8          if(n>=10&&n<=100) {
9              goto nhap;
10         }
11     }
```