

制御文 (その1)

山本昌志*

2004年6月25日

1 条件分岐文

分岐には、if文とswitch文があります。switch文はプログラムの構造が分かり難くなりますので、できるだけif文を使いましょう。

1.1 if文

1.1.1 ifのみの場合

制御式 (条件) が真の場合には特定の文を実行しますが、偽の場合には実行する文がありません。

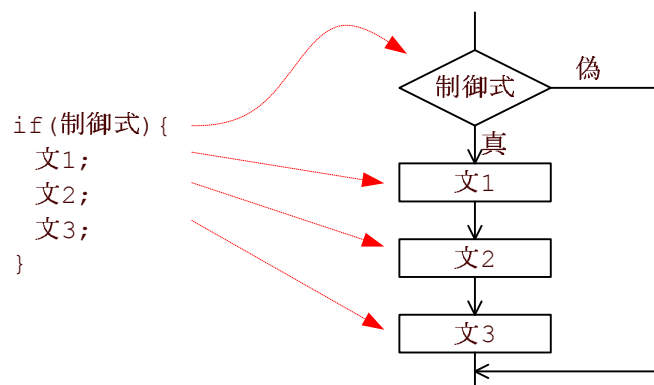


図 1: ifのみの場合

1.1.2 if~elseの場合

制御式 (条件) が真偽に応じて、実行する文が異なります。

* 国立秋田工業高等専門学校 電気情報工学科

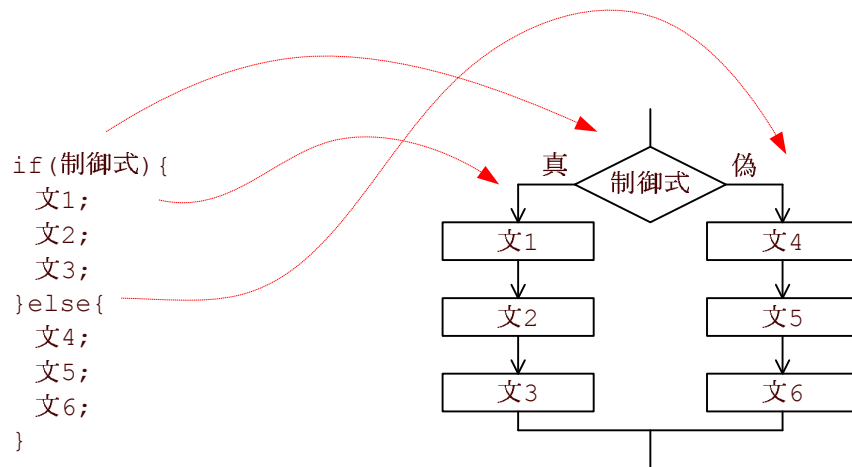


図 2: if ~ else の場合

1.1.3 if ~ else if ~ else の場合

制御式 (条件) が多段階になっています。最初に真にマッチしたブロックを実行します。制御式が真にならない場合は、else のブロックを実行します。

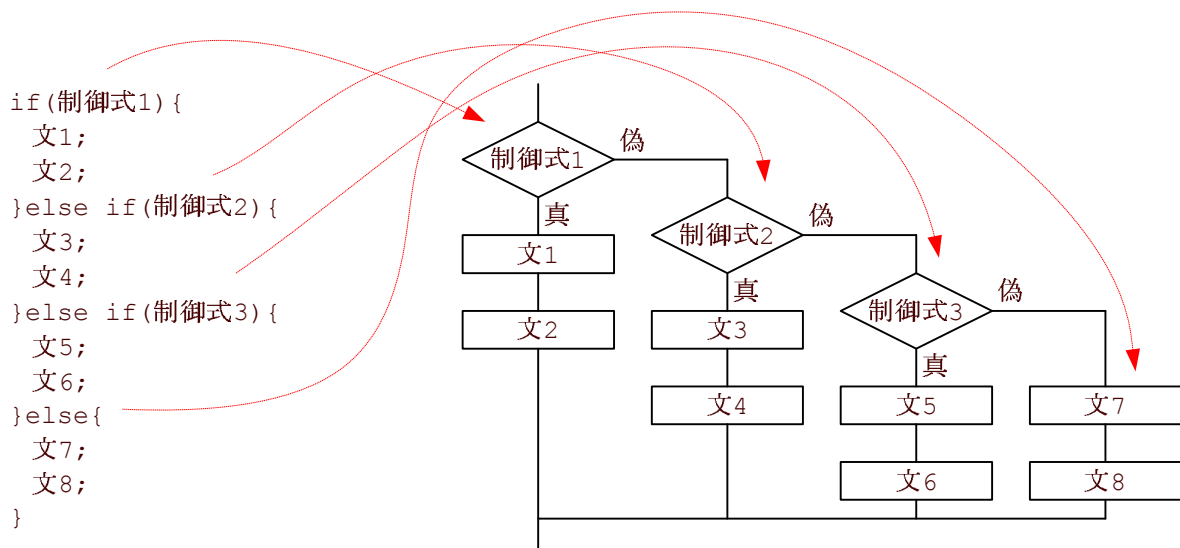


図 3: if ~ else if ~ else の場合

1.2 switch 文

case の使い方をまとめると、以下のようになる。

- 式は整数が結果とならなくてならない。ただの整数変数でも良い。整数変数 1 個でも C 言語は式と考える (数学と一緒に)。
- 式を評価して、その値がどれかの case 定数式と一致したら、その case ラベルにジャンプして、そこから処理を行う。
- 定数式には、整数を書く。浮動小数点数や変数、文字列は使えません。
- 処理は、break 文に出会うか、switch 文の終わりの } に出会うと終わる。
- 一般的には、各 case ごとに break 文を書く。しかし、書かなくても良い。
- 式の値が、どの case の定数式とも一致しなければ、default の文が実行される。default がなければ、どれも実行されない。
- case は、単純なラベルである。要するに、目印で実行文ではない。
- ひとつの実行グループに複数の case をつけても良い
- 同じ値をもつ 2 つの case を用いてはならない。

2 制御式

2.1 関係演算子

if 文の制御式は、論理演算を行います。それには、表 2 のような関係演算子と場合によっては論理演算しを使います。

- 関係演算子は、それを挟んでいる 2 個の数値の大小関係や、等値関係を判断します。判断の結果は、

正しいとき	1
間違っているとき	0

を返します。
- 演算子の左右の値が等しいか否かを調べるときは、==を使います。=ではありません。

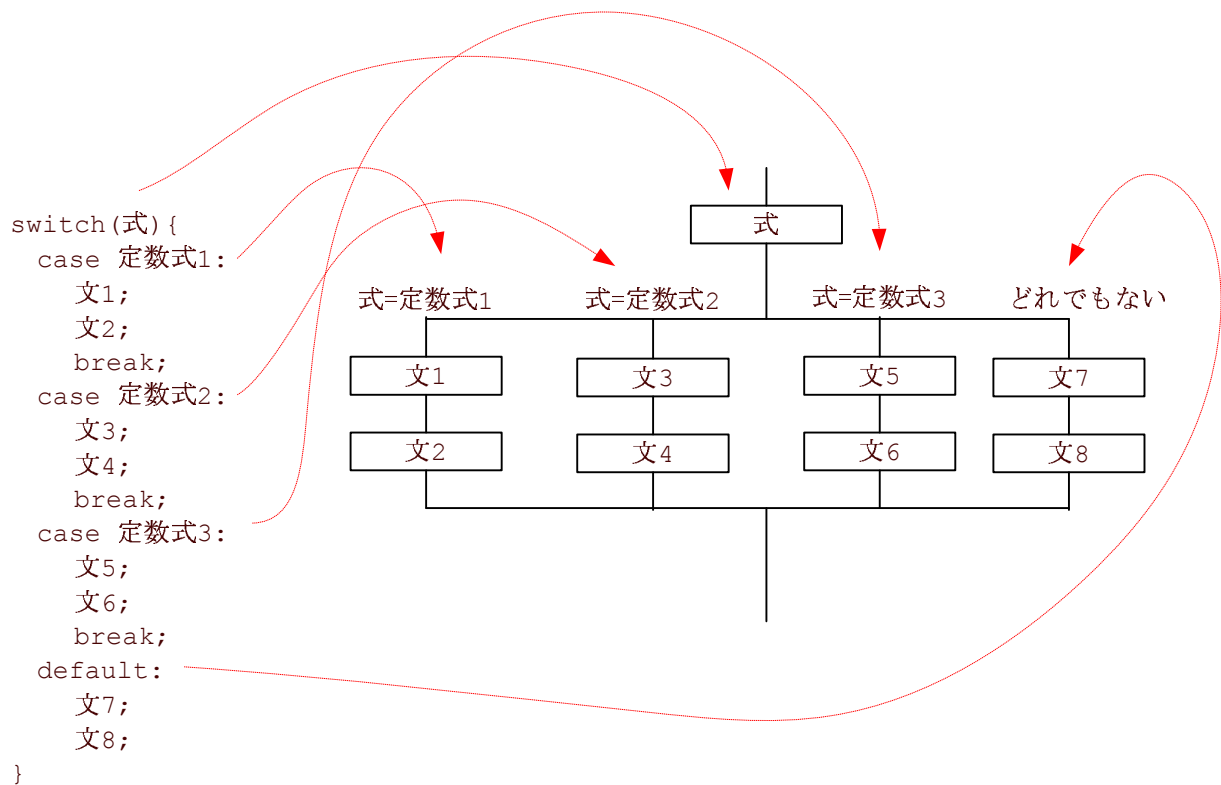


図 4: switch の場合

表 1: 関係演算子

演算子	機能	例
<	大小関係 (等号含まず)	if(a < b)
<=	大小関係 (等号服務)	if(a <= b)
>	大小関係 (等号含まず)	if(a > b)
>=	大小関係 (等号服務)	if(a >= b)
==	等しい	if(a == b)
!=	等しくない	if(a != b)

2.2 論理演算子

- 論理演算子は、ブール代数演算を行います。演算の結果、偽の場合は 0、真の場合は 1 を返します。
- C の場合、偽は 0 で、真は非 0 です。したがって、非 0 の整数の否定は 0 になります。

表 2: 關係演算子

演算子	機能	例
!	否定	if(!a)
&&	論理積 and	if(a && b)
	論理和 or	if(a b)