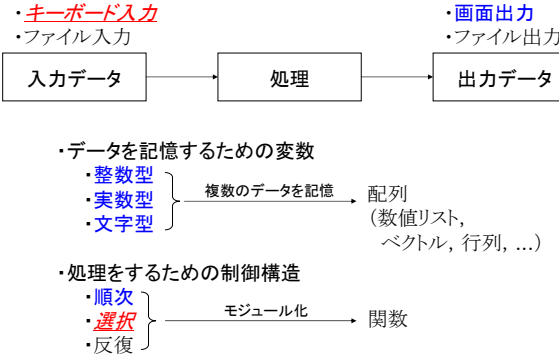


C言語を用いたプログラミング3

補足資料

- ・キーボード入力
- ・選択
 - ・if ~ else ~
 - ・switch ~ case ~
- ・中断
 - ・break

プログラム

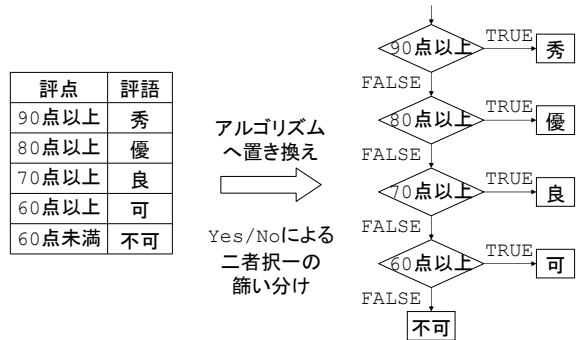


主な制御構造

制御構造	意味	分類
<u>if ~ else ~</u>	<u>条件分岐</u>	<u>選択</u>
<u>switch ~ case ~</u>	<u>多岐選択</u>	<u>selection</u>
while ~	前判定	反復 repetition
do ~ while	後判定	
for ~	繰り返す	
<u>break</u>	<u>処理の中断</u>	ループ処理
continue	処理のスキップ	

順次: concatenation

条件分岐 (if ~ else ~)



(用語) 篩: ふるい, sieve

関係演算子

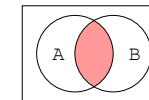
関係演算子	意味	使用例	演算結果	
			真の場合	偽の場合
==	等しい	a==b	1	0
!=	等しくない	a!=b	(!0)	
>=	以上	a>=b		
<=	以下	a<=b		
>	より大	a>b		
<	より小	a<b		

プログラム3.3参照

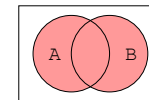
論理演算子 (プログラム3.2参照)

論理演算子	論理	使用例	演算結果	
			真の場合	偽の場合
&&	論理積 AND (かつ)	x>10 && x>100	1 (!0)	0
	論理和 OR (または)	x==1 x==2		
!	否定 NOT (~ではない)	!(x==1)		

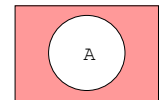
論理積 &&



論理和 ||



否定 !

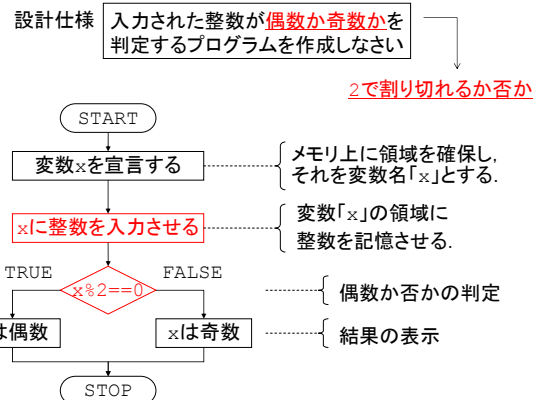


FAQ (論理演算子)

Q	0<=x<=□ という表記はだめですか？
A	正しい結果が得られません。なぜなら、偽を0, 真を1に置き換えるからです。例えば, (20>=15)>=10) は 数学的には真 ですが, C言語では偽 と判定されます。その処理手順は次のようになります。 ((20>=15)>=10) ⇒ ((真)>=10) ⇒ (1>=10) ⇒ 偽 必ず, 次のように複合条件を用いて下さい。 0<=x && x<=□

FAQ: Frequently Asked Question

プログラム3.4 (if ~ else ~)



算術演算子 (2.6節参照)

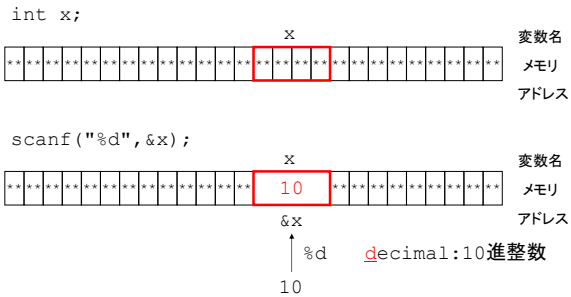
算術演算子	意味	整数同士	実数を含む場合
+	和	整数	実数
-	差	整数	実数
*	積	整数	実数
/	商	整数 (小数点以下切捨て)	実数
%	余り	整数	

11/2 → 5 余り 1

演算子の優先度	演算子	結合規則
高	()	→
↓	* / %	→
低	+ -	→

プログラム3.4の補足

10

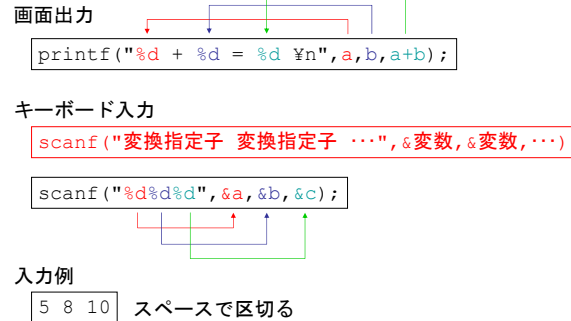


scanf: scan formatted input (書式付読み込み)

- scanf() 関数に表示機能は無い。
- 入力促進メッセージ(prompt)にはprintf() 関数を用いる。

複数の数値を読み込む

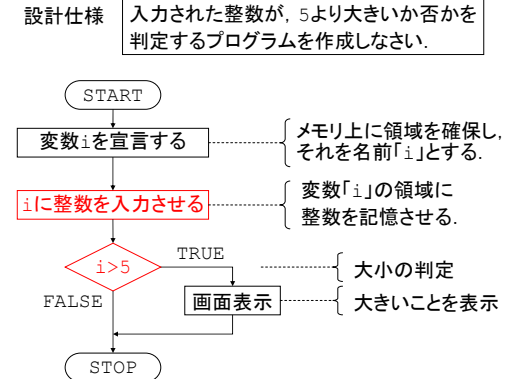
11



(注) 上記のように、複数入力が可能であるが、入力時に混乱が生じ易い。通常は、一つずつ入力させる。

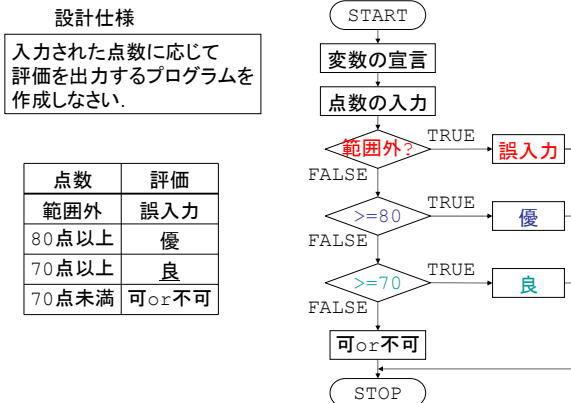
プログラム3.1 (if ~)

12



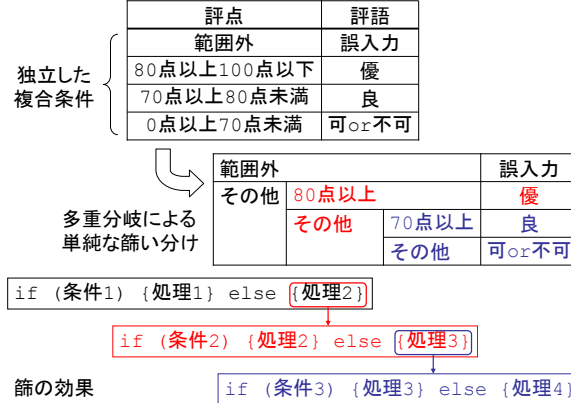
プログラム3.5(多重分岐)

13



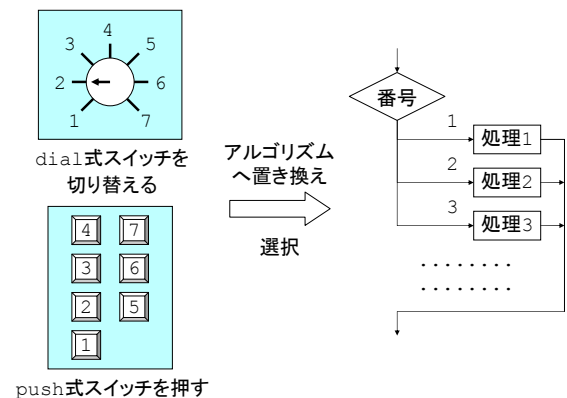
多重分岐の構造(プログラム3.5)

14



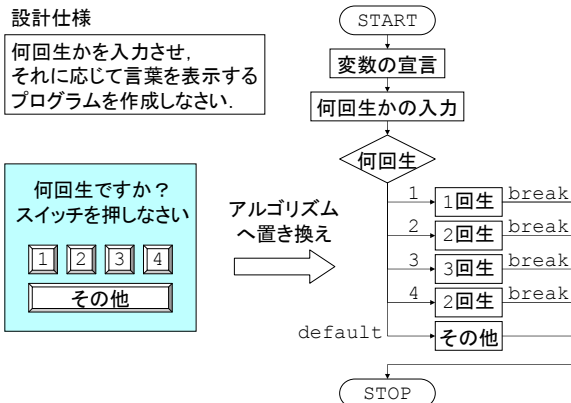
多岐選択 (switch ~ case ~, break)

15



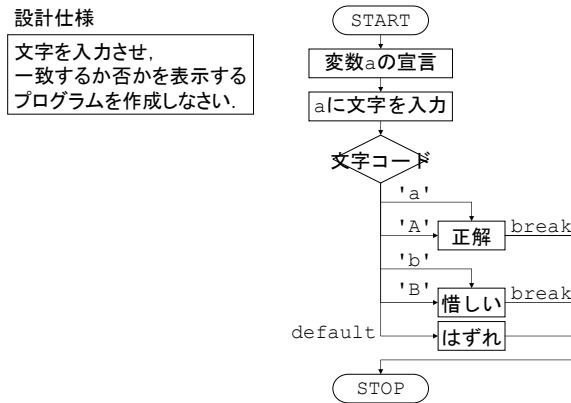
プログラム3.7

16



プログラム3.8

17



if - else と switch - case

18

