

2-3 集計

データ処理で行われる繰り返し処理の代表に集計処理がある。集計処理とは、複数のデータを加算して結果を求めることで、一般的には次のような式が繰り返される。

$$\text{集計変数} \leftarrow \text{集計変数} + \text{加算項目 (変数, 定数)}$$

このとき、注意することは、集計変数の初期値設定である。変数は、宣言しただけでは値が定義されない。したがって、初期値を設定しないと、この演算式の右辺にある集計変数の値が不定となり演算ができなくなってしまうからである。一般的には、集計変数には0が初期値として設定されるが、処理により例外もあるので注意が必要である。

カウンタと同様に、変数を集計に利用するためにも特別な定義（宣言）は必要ない。そのため、アルゴリズム中で、集計用の変数とカウンタ用の変数の区別がつきにくくなることがあるため、注意が必要である。一般的には、集計用の変数には「sum」や「kei」などという変数名を付けることが多い。

図 2.3.1 は、2を3回加算するアルゴリズムと、そのトレース結果表である。集計変数 sum とカウンタ変数 cnt の働きの違いに注意して、結果を確認してほしい。

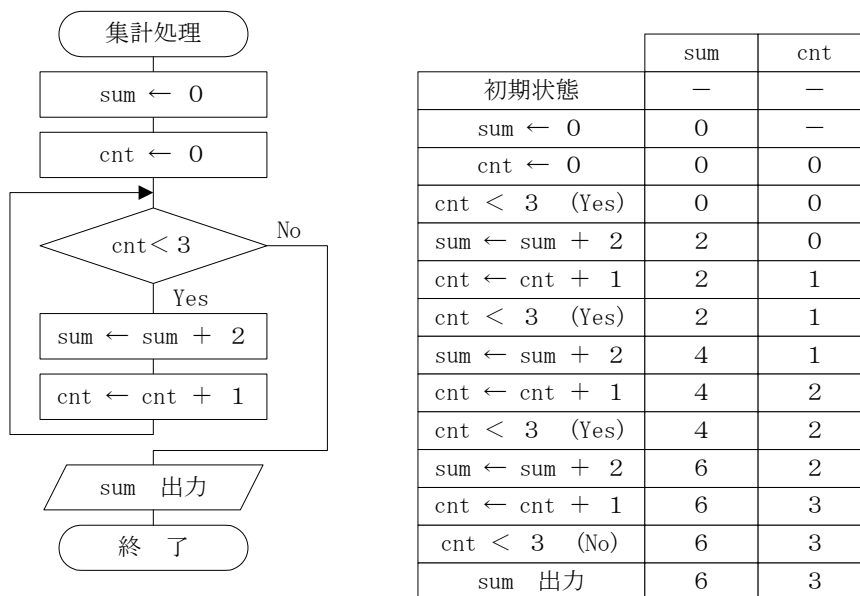
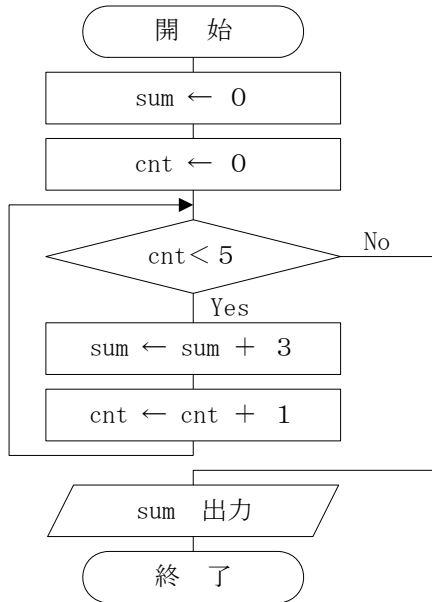


図 2.3.1 集計処理のアルゴリズム

問 2-16

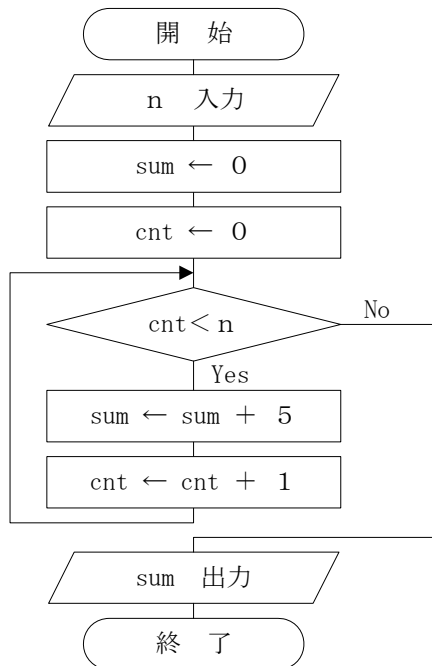
各フローチャートを実行した結果を答えよ。

(1)



《実行結果》

(2)



《実行結果 1》

2 (←入力)

《実行結果 2》

9 (←入力)

加算項目が定数の場合、集計結果は「定数×繰返し回数」となる。つまり、わざわざ集計処理にしなくても、単純な演算で結果を求められる。例えば、2を3回加算するアルゴリズムは、図 2.3.2 のように「 2×3 」という演算に書き換えることができる。

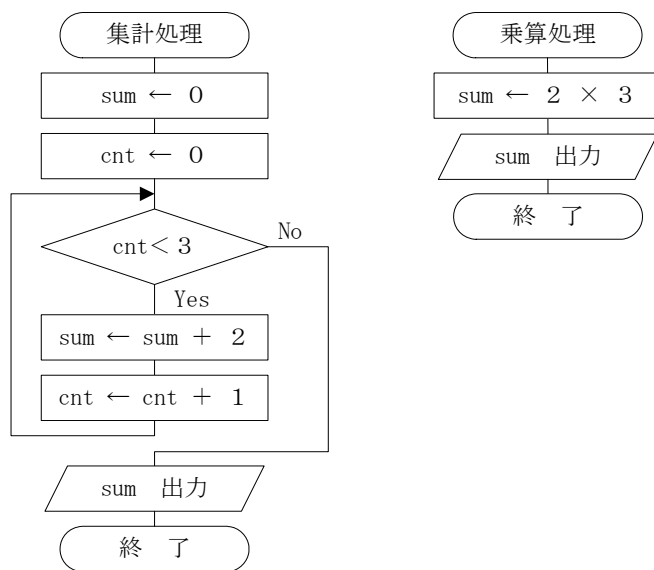


図 2.3.2 集計処理の書換え

コンピュータ内部には加算回路しかないため、乗算を加算の繰返し（集計処理）で処理することになる。したがって、定数を集計するアルゴリズム自体を理解しておくことは大切である。しかし、問題解決のアルゴリズムを考える場合には、定数の集計処理を表記する必要はあまりない。

基本的に集計処理は、加算項目の値が変化する場合に使用されることが多い。加算項目の値の変化には、ある一定の規則がある場合と、集計処理ごとに随時発生する（不規則に変化する）場合がある。

①加算項目が規則的に変化する場合

加算項目を規則的に変化させる例としては、「1～5の整数の和」を求める処理などがある。この処理は、加算項目を1, 2, 3, 4, 5と+1ずつ順番に変化させる。これは加算項目を変数とし、初期値1から繰り返しごとに+1ずつ加算することで実現できる。また、集計変数への加算は5回行われるため、この繰り返し制御をカウンタにより行う。このようにして作成したものが、図2.3.3の集計処理Aである。

ここで、集計処理Aをよく見ると、加算項目とカウンタの動きが非常によく似ていることに気づく。そこで、カウンタによる繰り返し制御を「0, 1, 2, 3, 4」の5回から「1, 2, 3, 4, 5」の5回に変えて加算項目として使用したのが集計処理Bである。

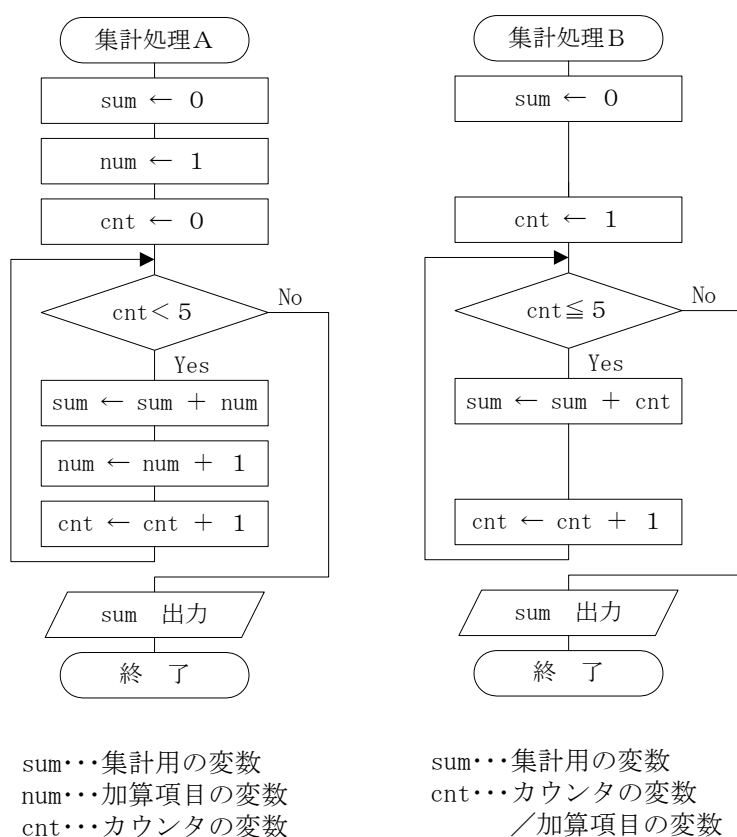


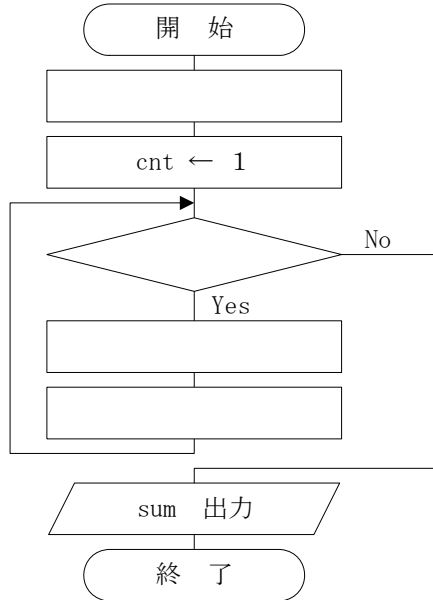
図 2.3.3 1～5の整数の和を求める

アルゴリズムとしては、カウンタと加算項目を別にしてある集計処理Aの方がよい。しかし、処理を少なくする目的などで、集計処理Bのようにカウンタを加算項目として利用する方が一般的に使用される。

問 2-17

処理の説明を読み、各フローチャートの空欄の処理を答えよ。

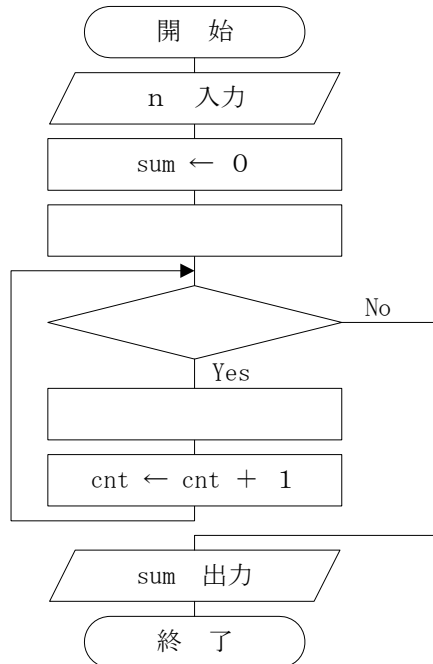
- (1) 1～10の整数の和を求める。



《実行結果》

55

- (2) ある数nを入力し、1～nの整数の和を求める。



《実行結果》

7 (←入力)
28 (←出力)