

1. フローチャートとは

フローチャートは、「流れ図」とも呼ばれます。

プログラムを作成する場合、次のような手順で作業を進めます。

- 1) 処理内容を、フローチャートで記述します。
 - 2) フローチャートに誤りが無いことを確認します。
 - 3) フローチャートに従って、プログラムを作成します。
 - ・プログラムが設計通りに動作している場合
プログラムの作成はここで終了です。
 - ・作成したプログラムが設計通りに動作しない場合
 - 4) 不具合が発生している箇所を探します。
 - 5) 正しい動作となるように、フローチャートを訂正します。
 - 6) フローチャートに従って、プログラムを作成します。
- 以後、不具合が無くなるまで4～6の作業を繰り返します。

2. フローチャート記号

ここでは、よく利用する記号を紹介します。

2.1 端子

処理の開始と終了を示します。半円と長方形が合体した図形です。

図形内に、開始または終了を意味する言葉を記述します。



開始や終了の語句は任意です。しかし、適当ではなく互いにペアとなる語句を採用します。

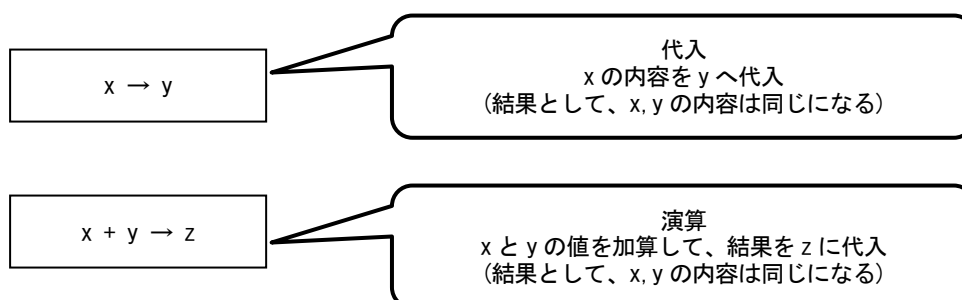
例：メインルーチン：START～END、START～STOP など。

サブルーチン：開始～戻る、START～RETURN など。

2.2 処理

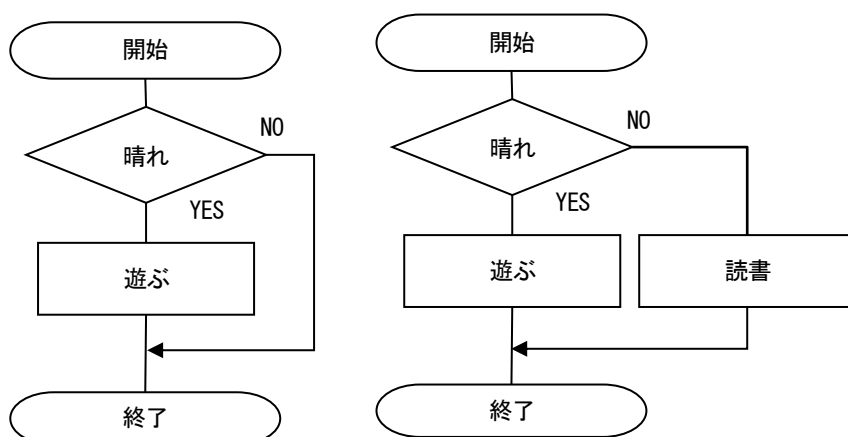
任意の処理内容を示します。長方形で表します。

図形内に、処理の内容を手短かに記述します。



2.3 判断

指定した条件に従って判断を行い、処理の流れを切り替えます。菱形で表現します。
図形内に判断条件、図形外に判断結果の流れを記述します。

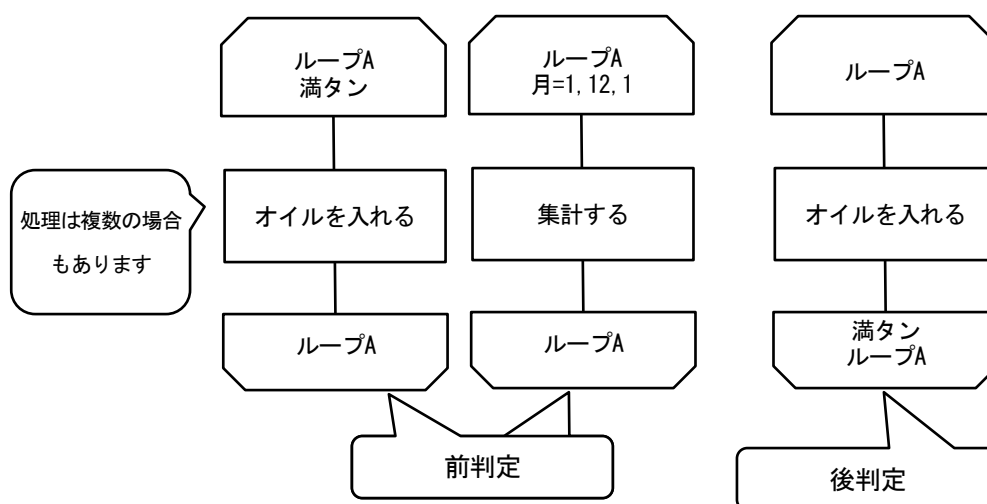


YES、NO の位置は任意ですが、できるだけ統一します。

2.4 ループ端

繰り返しの処理(ループ)を表します

次のように、2種類の6角形を必ずペアで記述します。



繰り返しの動作は、適切なタイミングで終了しなければいけません。

この終了するためのタイミングを決める条件を、「終了条件」といいます。

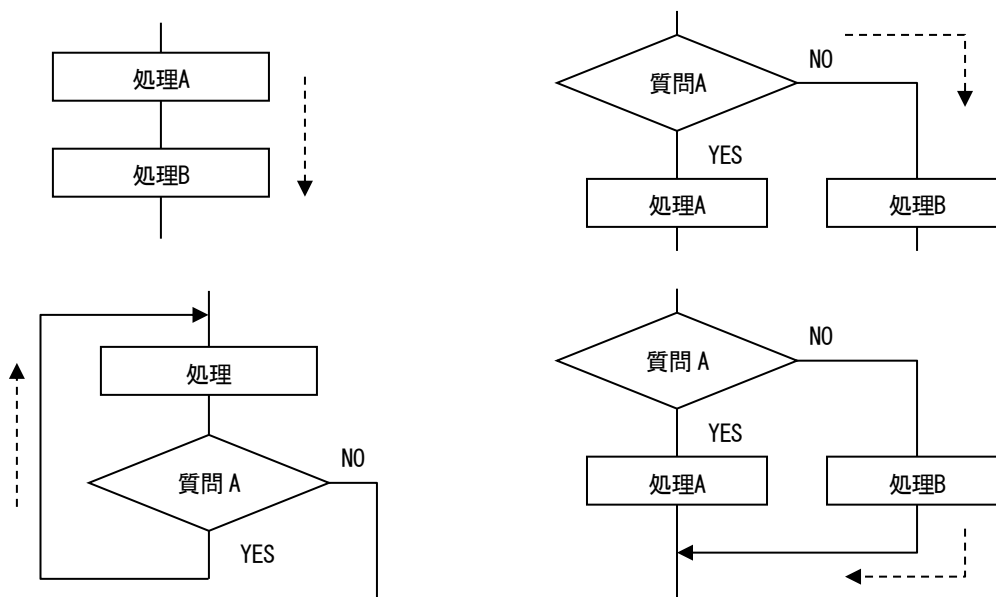
終了条件が無い場合、処理は無限に繰り返します。

※終了条件は、ビジネス系の処理では通常は指定します。しかし、組込み系では指定しないことが多くあります。

2.5 流れ線

処理の流れを表します。

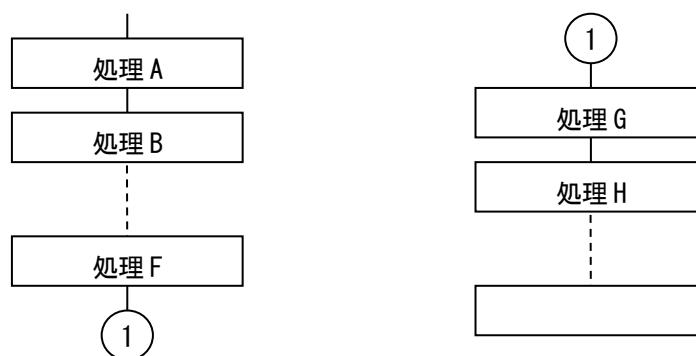
矢印は、流れが左から右または下から上に流れる場合に付けます。



2.6 結合子

フローチャートが1枚の用紙に収まらない場合や、流れ線が交差する場合などに利用します。

記号内には英数字を利用し、繋がる相手と同じ英数字を記述します。



2.7 媒体

ハードディスク、プリンター、キーボード、ディスプレイなどを表します。



3. モジュール分割とフローチャート

複雑なプログラムをフローチャートで表現する場合、全ての部分を詳細に記述していたのでは、却って理解しづらくなります。

そこで、あるまとまった単位ごとに分割してフローチャートを記述します。この一つ一つをモジュールといいます。そして、モジュールに分割されたプログラムを実際に作成する場合は、このモジュールごとに分けて制作します。

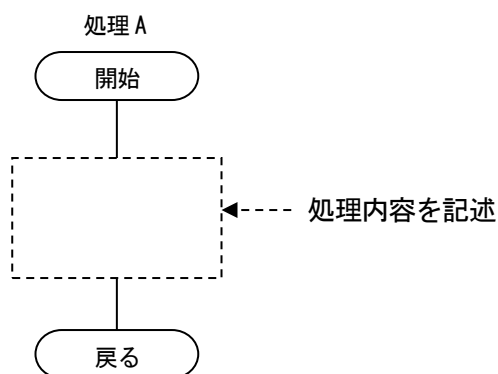
副定義(サブルーチン)



上位のモジュールを記述



下位のモジュールを記述



以上