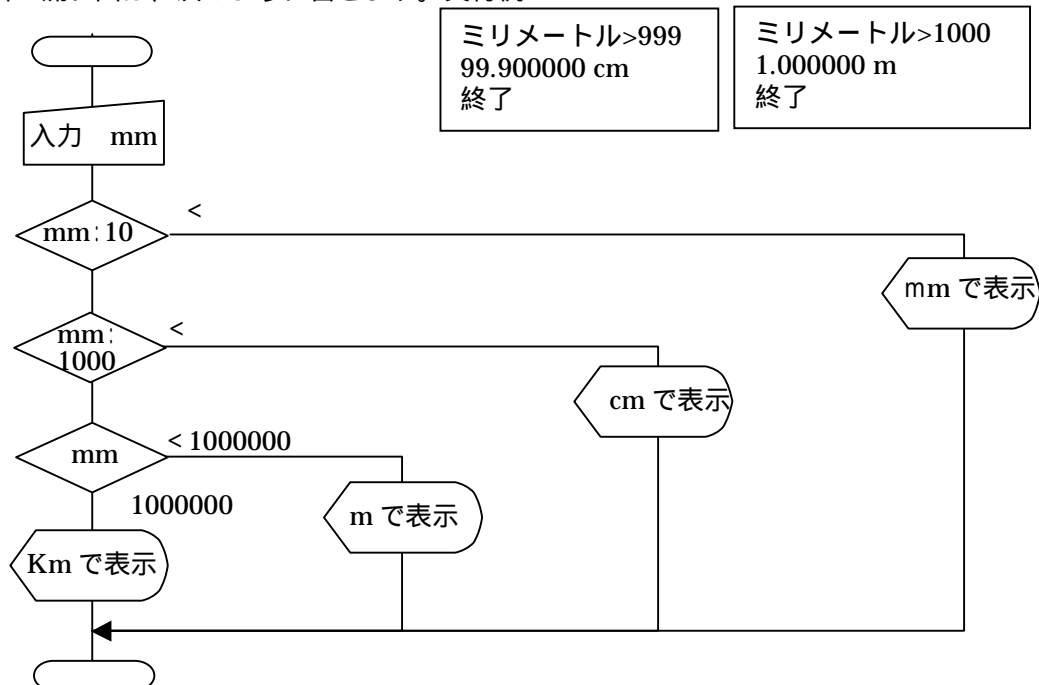


if else if else …if else 文

長さが mm(ミリ)を入力して、10mm 以上なら cm の単位で表示し、1000mm 以上なら m の単位で表示し、1000000mm 以上なら Km の単位で表示させる。

この場合の流れ図は、次のように書きます。実行例



このプログラムは、次のように作れました。

```

#include <stdio.h>
main()
{
    double mm; /* ミリメートル記憶 */
    printf("ミリメートル>");
    scanf("%lf" , & mm );

    if(mm < 10){
        printf("%f mm¥n", mm);
    } else {
        if(mm < 1000){
            printf("%f cm¥n", mm / 10);
        } else {
            if(mm < 1000000){
                printf("%f m¥n", mm / 1000);
            } else {
                printf("%f km¥n", mm / 1000000);
            }
        }
    }
    printf("終了¥n");
}
    
```

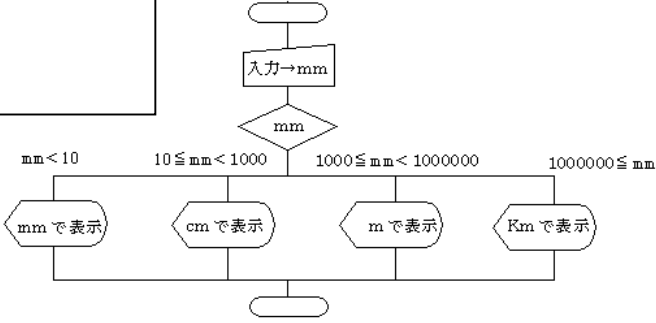
これは、正しい書き方の一つです。しかし、この書き方をすると、どんどん右へのインデントが大きくなり、読み難くなります。このような場合は、{ } を無しにして、インデントを付けずに、次のように書きます。その方が読みやすいプログラムになります。

```
#include <stdio.h>
main()
{
    double mm;      /* ミリメートル記憶 */
    printf("ミリメートル>");
    scanf("%lf" , & mm );

    if(mm < 10){
        printf("%f mm¥n", mm);
    } else if(mm < 1000){
        printf("%f cm¥n", mm / 10);
    } else if(mm < 1000000){
        printf("%f m¥n", mm / 1000);
    } else {
        printf("%f km¥n", mm / 1000000);
    }
    printf("終了");
}
```

前述のプログラムも、そう
ですが、比較する順番を
書き間違えると、
全く違う結果になってしま
います。
つまり、比較する順番が
ミソです。

このような書き方は、
前述で示した流れ図を、右のようにも
書けるということと、似ています。



問題1 上記プログラムをまず作成しましょう。正しく動いたら、do while でなく while で
入力が負になるまで繰り返しさせてみましょう。(負の入力では、入力したデータを表示させない
で、終了の表示だけ行うこととする。 tB01.c の名前で作る)

問題2 遊園地の個人の入場料が、18歳以上で3000円、13歳以上18歳未満で2000円、6歳
以上13歳未満で1000円、6歳未満で500円になっている。但し、団体人数として100人以上で
入る場合、個人入場より1割安くなる。同一年齢で入場するものとして、人数と年齢を入力す
ると、入場料の総額を表示するプログラムを作成する。終わったら、人数入力まで繰
り返すよう変更しなさい。(do while を使うか、while を使うかは自由です。tB02.c)

繰り返し処理追加前の実行結果例を示す。

人数>1
年齢>13
総額 2000 円です。

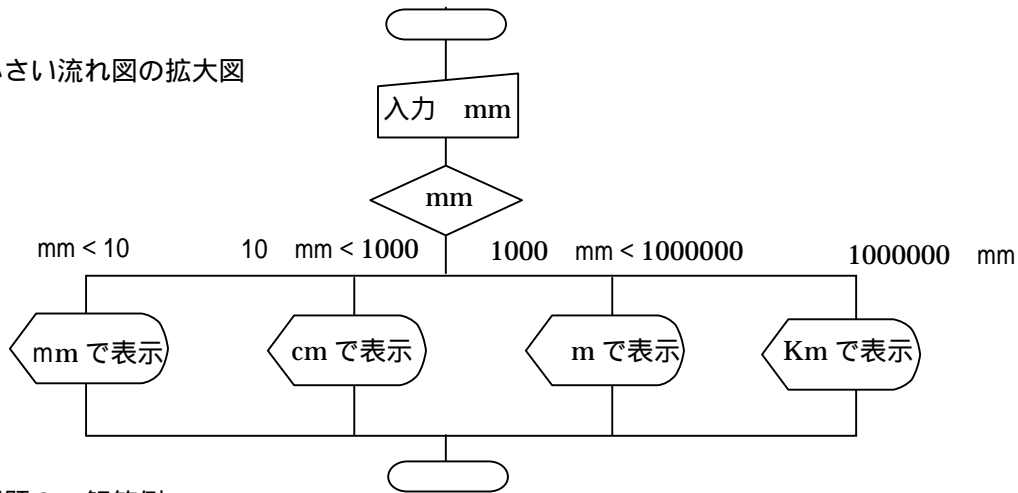
人数>100
年齢>13
総額 180000 円です。

人数>100
年齢>12
総額 90000 円です。

問題3 while の繰り返し練習問題 (tB03.c) 入力したデータの各桁の数値を列挙する。
何桁でもできるようにできるが、わからない場合は6桁限定入力で行う。

入力>123456
100000 の桁は、1 で 10000 の桁は、2 で 1000 の桁は、3 で 100 の桁は、4 で 10 の桁は、5 で
1 の桁は、6 です

小さい流れ図の拡大図



問題3 解答例

```

#include <stdio.h>
main()
{
    int data;
    int keta = 100000;
    printf("入力>");
    scanf("%d", &data);

    while(keta > 0){
        if(data >= 100000){
            printf("%dの桁は、%dで ", keta, data / 100000);
            data = data % 100000;
        } else if(data >= 10000){
            printf("%dの桁は、%dで ", keta, data / 10000);
            data = data % 10000;
        } else if(data >= 1000){
            printf("%dの桁は、%dで ", keta, data / 1000);
            data = data % 1000;
        } else if(data >= 100){
            printf("%dの桁は、%dで ", keta, data / 100);
            data = data % 100;
        } else if(data >= 10){
            printf("%dの桁は、%dで ", keta, data / 10);
            data = data % 10;
        } else {
            printf("%dの桁は、%dで ", keta, data / 1);
            data = data % 1;
        }
        keta = keta / 10;
    }
    printf("すゑ\n");
}
    
```

かく桁を、上から順番に処理する繰り返し

100000、10000、1000、100、10、1と変化するこれに、変数 keta を使えば、繰り返しの中で if は不要になる。

```

while(keta > 0){
    printf("%dの桁は、%dで ", keta, data / keta);
    data = data % keta;
    keta = keta / 10;
}
    
```

よって、この繰り返しの前に、keta を適切に求めれば、任意の桁に対応できる。