

[伝票番号<WAのとき。(同じ伝票番号)]

データの重複・混入，伝票番号のパンチミス，以前のデータのエラーなどが考えられる。

なお，エラー発生時には以後の処理のため，WAの値を修正しておく。

④ [伝票番号<最後の伝票番号のとき。]

データの不足，エンドカードが途中に入っている，バッチコントロールカードのパンチミスなどが考えられる。

(3) 得意先コードおよび商品コードのチェック

① チェックデジットによるチェック

② チェックデジットの算出

(例)

○コードの各けた (コード) 3 1 3 4
 $\times \times \times \times$
 にウェイトを乗 (ウェイト) 4 3 2 1
 ずる。 ↓↓↓↓

○乗算結果を加える。 $12+3+6+4=25$

○計25をモジュラス11で割る。

$$25 \div 11 = 2 \text{ 余り } 3$$

○モジュラスから余り3を引く。答をチェックデジットとする。 $11-3=8$

○減算結果8をコードに附加して，31348 とする。

③ チェック方法

コードを入力したら，上記算出方法により，チェックデジットを算出し，これと入力したチェックデジットが一致するかどうかを調べる。

(注) モジュラスとウェイトのとり方によってエラー検出率が異なる。つぎの表は，松下通信工業，植山喬氏の調査によるものである。

各種チェックデジットの法則の評価(%)

ケタ数	モジュラス	ウェイト	誤字
6	10	1. 2. 1. 2. 1. 2	94.3
6	10	1. 3. 1. 3. 1. 3	100.0
6	10	6. 5. 4. 3. 2. 1	87.0
6	10	8. 7. 4. 3. 2. 1	94.3
6	10	1. 3. 7. 1. 3. 7	100.0
6	11	6. 5. 4. 3. 2. 1	100.0

転 エ ラ ー	倒 エ ラ ー	複 エ ラ ー	ランダム エ ラ ー
100.0	0.0	90.0	90.0
95.0	0.0	90.0	90.0
100.0	96.0	90.0	90.0
100.0	96.0	90.0	90.0
95.0	96.0	90.0	90.0
100.0	100.0	90.9	90.9

② コードテーブルとの照合

プログラムに得意先コード，商品コードのテーブルを用意し，入力したコードがテーブル内のコードと一致するかどうかを調べる。

また，マスターファイルのコードとマッチングする方法もあるが，それには，変動ファイルをコード順に分類しておく処理が必要である。コードの変更・追加が頻繁に行われる場合などは，この方法による。

(4) 数量・単価・金額のチェック

○入力した金額が数量×単価の演算結果と一致するかどうかを調べる。不一致のときは，それぞれの項目についてパンチミス，手計算ミス，転記ミス等を疑ってみる。

○数量・単価は，過去の実績やその商品の性質から，ある範囲内の値をとることがわかっているならば，その妥当性についてチェックを行う。

○単価に円未満の端数がある場合には，求めた金額の端数処理について明確にしておかなければならない。

(5) 年・月・日のチェック

○日付については，それぞれとり得る最大値と最小値があるので，これと比較することによってエラーを検出する。

○コンソールから入力した年・月・日と比較して，処理対象範囲の期間内かどうかを調べる。

おわりに

これまで，種々のチェック方法について述べてきたが，このようなチェック方法をすべて行うものとは限らない。チェックに要する費用や時間等を考慮して，ある程度のチェックですませる場合もあろう。

エラー防止，エラー回復については，むしろ多くの問題点を含んでいる。

近年，これらの問題に加え，ともすると軽視されがちであった「人間」の問題も含めたチェックシステムが工夫されている。

参考文献

小沢 暢 夫著 インプットシステム設計

日刊工業