

OA用簡易言語Macro-Cの 表操作アクティビティ

7D-1

野寄雅人

常葉学園浜松大学

1. はじめに

Macro-Cはパソコンを使うエンドユーザにも良質のOA用アプリケーションが作成できるようなプログラム開発環境を提供することを目的とした簡易言語である。可能な限り非手続き部を増やし、残る手続き部においても、高機能ながら汎用性を損なわず、しかも内部処理が完全に隠蔽された部品を提供してプログラミングのウェイトを大幅に軽減している。

Macro-Cにおいて中心的な役割を果たすものが“アクティビティ”である。このアクティビティはインストラクションの指示によって忠実に行動する。インストラクションの中にアクティビティが実行すべきプロシージャを指示できるので、アクティビティは汎用性を損なわずに高度にマクロ化された部品とみることができる。

図1はMacro-Cによって作成した販売管理システムの伝票入力画面である。このような画面を使ったデータ入力処理などが容易に実現できる。図において、区、コード、品名などの項目名の書かれている行より下が表領域であり、この領域は上下左右にスクロールする。

2. 表領域の設定

表操作を行うアクティビティとしては、画面上のカード内へのデータ入力を行うINPUT_C、画面上のシールドシート内へのデータ入力を行うINPUT_SHT、ファイルからデータを読み込み画面上の表領域に表形式で表示するREAD_Tなどがある。これらはいずれも画面を使って行動するが、その画面形式は画面形式設定用エディタを用いて作成する。

図2に画面形式設定用エディタで作成した画面形式の例を示す。3行目以降がプログラム実行時にこの形で表示される。zzzz, 999, xxxxxxなどはスクリーンフィールドの書式であり、4つの*を角とする長方形の領域が画面上の表領域である。表領域の中、n, 9, 999などの書式の書かれている行より下が上下にスク

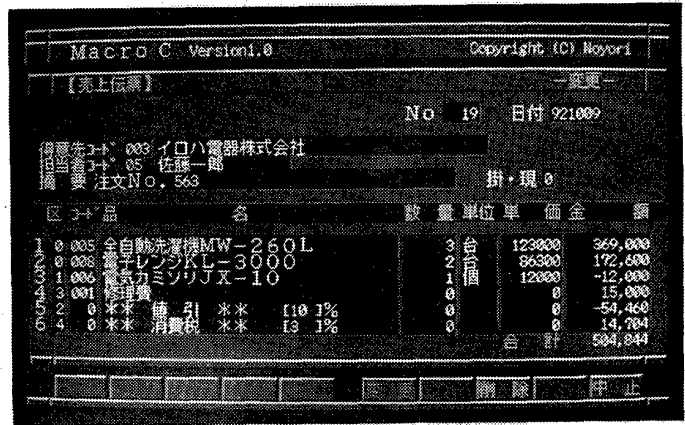


図1 売上伝票入力画面

ロールする領域である。また表領域の中、上下に位置する+より右側が左右にスクロールする領域である。

3. 表操作アクティビティとインストラクション

Macro-Cのアクティビティは与えられたインストラクションに従って忠実に行動する。INPUT_CとINPUT_SHTはウィンドウインストラクションに、READ_Tはファイルインストラクションに従う。

図3はINPUT_Cに対するウィンドウインストラクションの例である。pre_procの欄には、Noのフィールドにカーソルが入る直前に実行するプロシージャ名を記入する。またonKey_proの欄には指定のキーが押されたときに実行されるプロシージャ名を記入する。crsr欄はカーソルの動く順を記入する。

以上は表領域のフィールドに対しても適用される。したがって、表領域の各行において例えば単価×数量の計算し金額欄に表示するといったことが可能となる。

4. 表操作アクティビティのふるまい

(1) INPUT_C

下記のように手続きを記述し、図2をINST_Aのインストラクションとすれば、まず画面下のf・6, f・10のファンクションボックスが“登録”, “終了”と表示される。

```
INPUT_C (INST_A)
F06登録:TOUROKU ()
```

```

WINDOW_FORM program_name=ha40 form_no=0
position=1,3 size=78,18 type=1 table_length=6
【売上伝票】
                                     - 入力 -
                                     5
                                     N o zzzz 日付 zzzzzz 7
得 意 先 コーﾄ ` 999 xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 7
担 当 者 コーﾄ ` 99  xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 7
摘 要 xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  掛・現 9 7
* 区 コーﾄ ` 品 名 +数 量 単 位 単 価 金 額 *備 考 15
n 9 999 xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx zzzzzz xxxx zzzzzzz z2,zzz,z29 xxxxxxxxxxxx27
                                     27
                                     27
                                     27
*                                     + * 27
                                     合 計 z2,zzz,z29 26
    
```

図2 画面形式の例

F10終了:

そのあと画面上のカードの中のフィールドにデータを入力することができるようになる。ファンクションキー「f・6」あるいは「f・10」を押すとこのアクティビティは処理を終了する。終了したあとの処理はF06登録:あるいはF10終了:のあとに記述されたステートメントに従う。

処理の過程でアクティビティはINST_Aの指示に従う。フィールド7の入力を終わるとプロセスPROC_Gの手続きが実行される。またフィールド2の入力時「f・6」を押すとPROC_1の手続きが実行される。表領域の中の金額欄の入力を終わると表領域は左にスクロールし備考欄の入力に移る。表領域の2行目以降の入力は1行目と同様に行われ、4行目の入力を終わると表領域は上にスクロールし5行目の入力に移る。

(2) READ_T

下記のように手続きを記述するとファイルインストラクションに記述された検索条件に合うものをファイルか

ら連続的に読み込み画面上の表中に表形式で表示する。

READ_T (INST_F)

ESC:

カーソルキー↓←↑→を押すと表領域が上下左右にスクロールする。ESCキーを押すと処理を終了する。

5. おわりに

本稿では表操作を行うアクティビティのうちINPUT_CとINPUT_Tについて概説した。このアクティビティを用いれば簡明な記述で画面上のカードの中で容易に表を扱うことが可能になる。そのほかのINPUT_SHTなどの表操作を行うアクティビティについても同様である。

各アクティビティの内部処理を実現するプログラムステップはテンプレートプログラムを使って自動的に生成されるが、生成されるプログラムの内部は完全に隠蔽されている(さわる必要がない)。このため、汎用性を損なわずに手続き部のウェイトが大幅に軽減される。

```

window_no 0
sfld_instruction
  No crsr msgNo fncNo pre_proc post_proc onKey_proc1 onKey_proc2 onKey_proc3
  0
  1 0
  2 1 0 F06:PROC_1
  3 2
  4 3 0 F06:PROC_2
  5 4
  6 5
  7 6 0 PROC_G PROC_G
  8 7 1 PROC_H PROC_1
  9 8 0 F06:PROC_3 PROC_J PROC_K
  10 9 PROC_D
  11 10 PROC_E
  12 11 PROC_F
  13 12
  14 13
message
  0 0..売掛 1..現金(0,24,4)
  1 0..売上 1..返品 2..値引 3..諸費用 4..消費税(0,24,4)
function
  0 F06参 照(6)
    
```

図3 ウィンドウインストラクションの例