

INILL\$ - Initialise Doubly Linked-List Index Block

The INILL\$ routine initialises a separate Index Block for a doubly linked-list.

1. Invocation

To initialise the separate Index Block for a doubly linked-list code:

```
CALL INILL$ USING ib entries
```

where *ib* is the Index Block to be initialised; and *entries* is a PIC 9(4) COMP variable, or literal, which contains the number of entries, *n*, in the Index Block. The Index Block is assumed to have this structure:

```
01      IB
02 FILLER OCCURS n
03      NEXTI      PIC 9(4) COMP      * INDEX OF NEXT ENTRY
03      PREVI      PIC 9(4) COMP      * INDEX OF PREVIOUS ENTRY
```

2. STOP Codes and Exception Conditions

No STOP codes are generated by INILL\$.

The following EXIT codes may be returned by INILL\$:

EXIT code	\$\$COND	Description
15908	08	Exception returned from SVC-61.

3. Programming Notes

INILL\$ is only available with GSM SP-41, or later.

INILL\$ sets the first (N-1) NEXTI entries to (N+1); and the Nth NEXTI entry to 0. INILL\$ sets the first PREVI entry to 0; and entries 2 to N to (N-1).

INILL\$ can only be used when the indexes of the entries the doubly link-list are held in a separate table. INILL\$ could be extended to assume to the next and previous indexes are the first two PIC 9(4) COMP fields in a more complex structure (in which case the length of each entry would have to be passed as an extra parameter). For example

```
01      IB
02 FILLER OCCURS n
03      NEXTI      PIC 9(4) COMP      * INDEX OF NEXT ENTRY
03      PREVI      PIC 9(4) COMP      * INDEX OF PREVIOUS ENTRY
03      LLENTY      PIC X(?)          * ACTUAL DATA IN LINKED LIST
```

INILL\$ could be further extended to assume to the first two fields in the doubly linked-list are the “next” and “previous” pointer, respectively. As above, the length of each entry would have to be passed as an extra parameter). For example:

INILL\$ - Initialise Doubly Linked-List Index Block

```
01      IB
      02 FILLER OCCURS n
          03      NEXTPTR      PIC PTR      * POINTER TO NEXT ENTRY
          03      PREVPTR      PIC PTR      * POINTER TO PREVIOUS ENTRY
          03      LLENTY      PIC X(?)      * ACTUAL DATA IN LINKED LIST
```

4. Examples

None.

5. Copy-Books

None.

6. See Also

None.