

MCR10130

Increase GTSS Memory Limits to allow Jovial Compiler to Run

Revision 1.0

Eric Swenson

December 7, 2022

1 Problem Description

It is not possible to run the Jovial or Algol compilers within the GTSS subsystem on Multics due to memory size limits being too small in the GTSS configuration. As a result, programs terminate with the following error:

```
m4/n4-i/o lim. call/save
*ABT 0003t-01 @1043.3 m4/n4-i/o lim. call/save
```

Increasing those limits allows the Jovial (and Algol) compilers to run.

Ticket <https://multics-trac.swenson.org/ticket/291> reports the failure to run the Jovial compiler.

2 Proposed Fix

The proposed fix is to simply increase the memory size limit to the point where the Jovial (and Algol) compilers now run. The new limit was determined through experimentation and gradual increasing until the smallest value that supported the Jovial compiler was found.

The limits are specified in the program `gcos_control_tables_.alm` using an ALM macro thus:

```
define  progra,?????,,i*,5,17,5,dollar_t
```

The 17 specifies the memory limit of 17 pages. Increasing this value to 27 pages allows the Jovial (and Algol) compilers to run.

A `compare_ascii` output showing the change is below:

```
cpa [lpn gcos_control_tables_.alm] ==

A277          define  progra,?????,,i*,5,17,5,dollar_t
Changed by B to:
```

```
B277          define   progra,?????,,i*,5,27,5,dollar_t
```

Comparison finished: 1 difference, 2 lines.

GHM 16:00 1.325 1 Swenson

3 Documentation

No changes to documentation are necessary – as the changes are only internal or have never been documented.

4 Test

First, the gtss subsystem will be used to invoke the compiler as specified in the ticket, making sure that it runs rather than aborts due to limits. Then a few other gtss runs will be performed to make sure that everything continues to work.

5 Version History

2022-12-07	1.0	Initial version of the MCR.
------------	-----	-----------------------------