

INTERNATIONAL MARITIME ORGANIZATION

4 ALBERT EMBANKMENT
LONDON SE1 7SR

Telephone: 0171-735 7611
Fax: 0171-587 3210
Telex: 23588 IMOLDN G



E

Ref. T1/3.03

MSC/Circ.854
12 June 1998

GUIDELINES FOR SHIPBOARD LOADING AND STABILITY COMPUTER PROGRAMS

1 The Maritime Safety Committee, at its sixty-ninth session (11 to 20 May 1998), approved Guidelines for shipboard loading and stability computer programs, set out in the annex, aiming at ensuring that as new programs are developed and introduced into service, they do not overlook potential vulnerabilities which can inadvertently cause human error problems. The Guidelines should be applied where computer-based systems are used to perform functions to assist operators in monitoring and verifying specific conditions and performance of the ship.

2 The Committee decided that the Guidelines should be reviewed in the future, on the basis of new technological developments and in the light of experience gained from their application.

3 Member Governments are invited to bring the Guidelines to the attention of interested parties as they deem appropriate.

ANNEX**GUIDELINES FOR SHIPBOARD LOADING AND STABILITY COMPUTER PROGRAMS****1 Scope**

These guidelines may be applied where computer-based systems are used to perform functions, such as:

- predicting draughts and trim and verifying that limiting stability parameters, such as "required GM_T" are met;
- tank instrumentation systems used to provide direct electronic input of liquid loads (cargo, fuel, ballast, etc.) into the computer, bypassing the human measurement and data entry steps;
- operators of containerships may want to verify that over-the-bow bridge visibility requirements are met;
- operators of chemical parcel tankers may want to integrate chemical compatibility data to create voyage-specific/cargo-specific loading plans, thereby optimizing cargo flexibility;
- similarly, OBO operators may want a system which can accommodate multiple bulk cargoes of different densities and compute bending stresses with more precision;
- a program system could be used to monitor real-time hull bending stresses during loading/discharging operations, or due to sea conditions while under way via sensor systems that provide direct input to the computer; and
- a calculating damage stability conditions integrating loading data and flooded compartment characteristics.

2 General requirements

2.1 *Units:* Basic stability calculations are performed using weights, typically Ltons or Mtons. However, some cargoes are more commonly measured in short tons, TEUs, or barrels. Other liquid loads (fuel and ballast) might be initially measured as soundings or ullages. The program developer may wish to make its program more convenient for the user to enter data in these alternate units. If so, the program should minimize chances for unit confusion and, wherever possible, weight conversions should be calculated by the computer. Screen displays and print-outs should then present both the entered value and the computational weight value side-by-side.

2.2 *Data and program protection:* Although the program should be flexible enough to allow the user to override default data, certain data, such as lightship characteristics, allowable bending stress, required GM, as well as the program itself, should be protected against user revision. This could be achieved by furnishing the ship with compiled or read-only versions.

2.3 *Back-up of data:* Copies of all constant data residing in computer files, such as ship geometry and tables, should be available on independent storage units, such as tape or floppy disks. The number of such copies should not be less than two.

3 User interface

3.1 *"Home" screen:* The program should have a simple command (keystroke/icon) that returns the user directly to a familiar "home" screen from any of the loading screens. This allows a "lost" user (who may have got disoriented among various loading screens) to quickly re-establish their orientation.

3.2 *"Help" functions:* The program should have easily-accessed "help" functions such as designated function keys, or an on-screen menu bar.

3.3 *Default loading:* A default loading condition should reflect any special loading or operating requirements imposed by the ship's stability booklet (such as locked-in ballast requirements).

3.4 *Input and output data screening:* The program should check data entered by the user for reasonableness in order to screen out possible input errors, for example, a cargo tank entry which exceeds the capacity of the tank. The program should not reject the entry as there may be special loading scenarios where unusual data must be entered, but it should clearly indicate to the user that the entry is out of expected bounds. Similarly, the program should alert the user if an output parameter such as "predicted GM" is out of expected bounds.

3.5 *Alerts:* The system should alert the user if an output indicates a critical, or possibly dangerous situation. Alerts should, when possible, be augmented by audio signals. It is recommended that the graphical presentation and audio signals are different in case of critical events and user errors.

3.6 *Extra loading entry lines:* In most cases, load entries will be of the fixed-location type where LCGs, VCGs, etc., are pre-displayed and the user only needs to enter a weight value. However, the program should include several extra blank lines to allow additional non-fixed load entries where the user can enter VCG, LCG, TCG, etc. Examples of non-fixed load entries might be an unusual deck cargo, temporary ballast or damaged stability calculations (where a flooded compartment could be entered as if it were a tank).

3.7 *Print-outs:* Each loading condition print-out should automatically contain the name of the ship and the date of print-out; user should be prompted to enter a title for the condition as well. This information should be repeated on each page of the print-out.

4 Training and documentation

4.1 *User training:* Training/tutorial material should be provided, as appropriate, for the sophistication of the program. This may range from formal classroom sessions to tutorial videotapes and/or self-study lesson plans.

4.2 *Documentation:* The software should be accompanied by a user's manual and a programmer's manual.

4.2.1 The user's manual should be written for the direct user (ship's officers) and should include the following elements:

- .1 *Identification:* the manual should have a unique identification number that matches an on-screen ID number in the program. It should also clearly identify the stability booklet from which the lightship data is taken.

- .2 *System requirements:* identifies computer system hardware and software requirements such as compatible computers, operating system, memory requirements and other special requirements, such as video graphics, mouse, printer, etc.
- .3 *File management:* a list of all relevant software files, giving name, size, date and a brief description of each. The manual should also explain how any user-generated files, such as saved loading conditions, are named. These measures should allow the user to review the disk directory and verify that the correct current files are present.
- .4 *Instructions:* a clear explanation of how to install, use, and troubleshoot the program. The instructions should be user-friendly, recognizing that the user is a ship officer.
- .5 *Information sources:* a list of all ship-specific plans, drawings, tables, other documents, etc., which provided information used in the program. In most cases, this information will probably come from the ship's approved stability booklet; however, other sources should be clearly identified. Ideally, all such information sources should themselves be annotated to the effect that they were used in developing the program (so that future revisions to the drawing will also prompt a review of the program).

4.2.2 *The programmer's manual* is not expected to be furnished to the ship; it is for use by select persons familiar with programming (but who may not necessarily be the original program writers) when it becomes necessary to revise the program as a consequence of changes to the ship. The programmer's manual should carefully document the program's workings, and include a flowchart and an annotated program listing. This manual should explain how to edit the program, especially to revise ship-specific data (lightship data, hydrostatic characteristics, weight and moment data, tank capacities, etc.).

4.3 *Program and documentation control:* A careful procedure should be established so that revisions to the program are properly tracked and forwarded to the ship. Each revision delivered to the ship should include change pages to the user's manual and instructions on how to delete obsolete files and install replacement (revised) files. The process should include an "action complete" report back to shoreside management.

5 Program functionality

Program functionality: A manner for independently verifying the program's functioning should be provided. Ideally, the opening screen (when the program is first brought up) should present a self-diagnostic report on program functioning. Alternatively, a range of sample loading conditions can be furnished (on paper) which can be manually entered into the program for comparison with correct draughts, trim and available GM. The sample conditions may be the same as found in the ship's approved stability booklet, or separate samples included in the program's user manual.