

Sault College of Applied Arts and Technology sault ste. marie

Course Outline

COMPUTER PROGRAMMING

- EDP 105-2

(3rd sem Forestry)
(4th sem. Geology)

General Objectives:

1. To provide a general introduction to the computer.
2. An knowledge of computer and programming terminology.
3. An understanding and useage of the macros in the programming language FORTRAN IV.
4. A detailed and critical outlook regarding logic and attention to detail.
5. An appreciation of computers; and a realization that they can be used in any field provided that the rules, logic and FORTRAN format are strictly followed.
6. To have each student formulate a task in his own technical area; organize it to computer specification and program the problem using the FORTRAN language.

COMPUTER PROGRAMMING

TOPIC NO.	PERIODS	TOPIC DESCRIPTION
1	20	<p><u>Flow Charting</u></p> <p>Symbols for diagramming(A.S.A.) Techniques Exercises</p>
2	6	<p><u>Basic Elements of Fortran</u></p> <p>Development of Fortran Types of statements Input Output Arithmetic Transfer of control Looping Compiler directing</p>
3	3	<p><u>Rules of the Coding Sheet</u></p> <p>Absolute accuracy in placement Neatness Letter and numerical distinction e.g. 0 (num.) Ø (alpha) Capitals only Usefulness of comments</p>
4	11	<p><u>Language</u></p> <p>Constants & variables Real constants & variables Integer constants & variables Exercises in distinguishing Data words in storage</p>

COMPUTER PROGRAMMING

TOPIC NO.	PERIODS	TOPIC DESCRIPTION
5	10	<u>Arithmetic Statements</u> Operation symbols Rules for formation of arithmetic statements Exercises in converting formula to FORTRAN statements & vice versa Determination of valid vs invalid arithmetic statements
6	15	<u>Basic Input-Output Statements</u> a. <u>Write statement</u> - general form - F - E - I Connection between I/O macro and FORMAT Truncation, positioning of data, right and left fill Exercises involving arithmetic statements and the output macro Output of arrays b. <u>Read Statement</u> Format statement Incorporate exercises involving use of the input macro Input of arrays Data statement
7	10	<u>Providing Descriptive Labels & Headings</u> General rules of printer layout sheet Positioning of data for informative output Fortran coding necessary to create required output Format - A, X, H, \', T Exercises
8	5	<u>Transfer of Control Statements</u> Unconditional GØ TØ Conditional GØ TØ

COMPUTER PROGRAMMING

TOPIC NO.	PERIODS	TOPIC DESCRIPTION
9	10	<u>IF Statement</u> General form Purpose Exercises involving all statements to date - follow through from flow chart to final execution
10	5	<u>Stop, Pause, End, Call Exit, statements</u> <u>STOP n</u> Purpose General form Placement <u>PAUSE n</u> Purpose General form Placement <u>END</u> Purpose Placement
11	40	<u>CASE STUDIES</u>

Specific

Objectives

To instruct the student in:

1. The general use of computers in industry
2. A brief history of computers.
3. The basic features of all computer systems.
4. The stored program concept.
5. The recognition & identification of all computer peripherals.
6. The concept of the binary system.
7. The reason for the binary system in all digital computers.
8. The meaning of FORTRAN and the existence of other computer languages.
9. The rules of the FORTRAN statement.
10. The FORTRAN character set.
11. The basic difference between variables and constants.
12. The meaning of the mode.
13. Identifying the difference between floating pt. (REAL) and fixed pt. (INTEGER) mode.
14. The identification of floating point constants, fltg. pt. variables. fixed point constants and fixed point variables.
15. The meaning and restrictions involved in the use of reserved words.
16. The meaning of MACRO and the types of MACROS involved in the language FORTRAN.
17. The mathematical notations of the arithmetic statement.

Continued

18. The rules of the arithmetic statement.
19. The use & abuse of the parentheses in arithmetic expressions.
20. The conversion of algebraic expressions to FORTRAN statements.
21. The hierarchy of mathematical operations.
22. The identifications & subsequent correction of FORTRAN arithmetic statement errors: ie order, formation, mode violation, variable name length or formation, or logic.
23. The necessity of mode conversion.
24. Analyzing the problem.
25. The meaning and need for flow charts.
26. The correct techniques and thought processes necessary to transfer logic into flow charts.
27. The rules of the FORTRAN coding sheet.
28. The use and placement of the WRITE statement.
29. The output FORMAT statement using F, I, E, T, X, H notation.
30. The analyzing of the WRITE and FORMAT combination with regard to desired results on the printed output page.
31. The coding of programs to incorporate all past MACROS and to include new ones.
32. Key punching rules and the transfer from coding sheet to punched card.
33. The order, need, and accuracy necessary in the preparation of J.C.L. (LONTA
CA)
34. The proper deck-set-up for the FORTRAN language.
35. The meaning of a compilation

36. The meaning of the execution phase.
37. The interpretation of the listing.
38. How to locate and correct compilation errors.
39. The testing of output feasibility ie- do I have what I want.
40. The use and placement of the READ macro.
41. The use of the input FORMAT statemtn considering modes and spacing.
42. The preparation of data cards.
43. The debugging and checking of the READ and input FORMAT statement with the data card format.
44. The meaning of and need for branching in a program.
45. The unconditional GO To macro.
46. The use of the statement number..
47. The meaning of undefined statement nos. on the listing.
48. Writing sample programs utilizing the branching techniques.
49. The meaning of the term looping.
50. The building of loops.
51. The format of the IF statement.
52. The meaning and set-up of F and I counters.
53. The use of counters in loop building.
54. Additional uses of counters in programming situations.
55. The need, placement and format of the CALL EXIT statement.
56. The use and placement of the ^{END} STGP statement.
57. The use and placement of comment cards.
58. The program structure involved in continuation lines.
59. The conditional GO TO statement in decision making.
60. The use and incorporation of the arithmetic IF statement in decision making.
61. Alphaeric and alphabetic notation for input and output formats.

62. How to set up headings, titles and subtitles using the output WRITE and FORMAT combination.
63. The meaning and use of carriage control characters to control vertical spacing and page shipping.
64. The availability of library functions to perform ALOG, SQRT, ABS, IABS, EXP, ATAN, COS, SIN.
65. The correct usage of library functions.
66. The reason for program truncation on output formats. *Printed*
67. The use of repetitive FORMAT specifications.
68. The technique of incrementation using counters.
69. The treatment of blank fields.
70. The meaning of fields, records and files.

SPECIFIC GROUPING

The following is a breakdown of the specific objectives stated previously. As the course progresses, you will note the shift of emphasis from one group to another. However, be cautioned that a shift to one does not mean an exclusion of the others. Knowledge of 1 group acts as a plateau to another and must be realized for success in the next level.

Group I.

The specific objectives stated below will not be tested as such, but are essential in the preparation and successful completion of programmes assigned.

These include:

- The FORTRAN character set.
- The identification and subsequent correction of FORTRAN arithmetic statement errors: i.e. order, formation, mode violation, variable name length or formation, or logic.
- Analyzing the problem.
- The meaning and need for flow charts.
- The correct techniques and thought processes necessary to transfer logic into flow charts.
- The analyzing of the WRITE and FORMAT combination with regard to desired results on the printed output page.
- The coding of programs to incorporate all past MACROS and to include new ones.
- Key punching rules and the transfer from coding sheet to punched card.
- The order, need, and accuracy necessary in the preparation of J.C.L.
- The proper deck-set-up for the FORTRAN language.
- The interpretation of the listing.
- How to locate and correct compilation errors.
- The testing of output feasibility: i.e. Do I have what I want.
- The preparation of data cards.
- The debugging and checking of the READ and input FORMAT statement with the data card format.
- The meaning of undefined statement nos. on the listing.
- Writing sample programs utilizing the branching techniques.
- The meaning of the term looping.
- The reason for program truncation on output formats.

Group II.

This group deals with the FORTRAN language rules. All objectives stated are essential in both the preparation of programmes and in the testing environment.

These include:

- The rules of the FORTRAN statement.
- The basic difference between variables and constants.
- The meaning of the mode.
- Identifying the difference between floating pt. (REAL) and fixed pt. (INTEGER) mode.
- The identification of floating point constants, fltg. pt. variables, fixed point constants and fixed point variables.
- The meaning and restrictions involved in the use of reserved words.
- The mathematical notations of the arithmetic statement.
- The rules of the arithmetic statement.
- The use and abuse of the parentheses in arithmetic expressions.
- The conversion of algebraic expressions to FORTRAN statements.
- The hierarchy of mathematical operations.
- The identification and subsequent correction of FORTRAN arithmetic statement errors: i.e. order, formation, mode violations, variable name length or formation, or logic.
- The necessity of mode conversion.
- Analyzing the problem.
- The rules of the FORTRAN coding sheet
- The use and placement of the WRITE statement.
- The output FORMAT statement using F, I, E, T, X, H notation.
- The analyzing of the WRITE and FORMAT combination with regard to desired results on the printed output page.
- The use and placement of the READ macro.
- The use of the input FORMAT statement considering modes and spacing.
- The meaning of and need for branching in a program.
- The unconditional GO To macro.
- The use of the statement number.
- The meaning of the term looping.
- The building of loops.
- The format of the IF statement.
- The meaning and set-up of F and I counters.
- The use of counters in loop building.

- Additional uses of counters in programming situations.
- The need, placement and format of the CALL EXIT statement.
- The use and placement of the END statement.
- The use and placement of comment cards.
- The program structure involved in continuation lines.
- The conditional GO TO statement in decision making.
- The use and incorporation of the arithmetic IF statement in decision making.
- Alphameric and alphabetic notation for input and output formats.
- How to set up headings, titles and subtitles using the output WRITE and FORMAT combination.
- The meaning and use of carriage control characters to control vertical spacing and page shipping.
- The availability of library functions to perform ALOG, SQRT, ABS, IABS, EXP, ATAN, COS, SIN.
- The correct usage of library functions.
- The reason for program truncation on output formats.
- The use of repetitive FORMAT specifications..
- The technique of incrementation using counters
- The treatment of blank fields.

Group III.

This final grouping includes objectives which are common to all computer systems and languages. These objectives will be foremost during the beginning phases of the course and will be used as test material. As groups 1 and 2 become prime, this group will become secondary and shall serve as reference material only.

These include:

- The general use of computers in industry.
- A brief history of computers.
- The basic features of all computer systems.
- The stored program concept.
- The recognition and identification of all computer peripherals.
- The concept of the binary system.
- The reason for the binary system in all digital computers.
- The meaning of FORTRAN and the existence of other computer languages.
- The meaning of a compilation.
- The meaning of the execution phase.