

# 2006 Assessment Criteria of Programme Dossier

Include these sheets with all future dossier submissions so sections already marked need not be remarked.

Date: \_\_\_\_\_ Author: \_\_\_\_\_ Phones: \_\_\_\_\_ - \_\_\_\_\_ - \_\_\_\_\_  
Home Cell

<b>Assigned IBO Dossier Mark = Mastery Factor [     ] X Assessment Evaluation [     ] =</b>	<b><u>      </u></b>	<b><u>/50</u></b>
[ CTS == Career & Technology Studies, IBO == International Baccalaureate Organization ]	<b>CTS ↓</b>	<b>IBO ↓</b>
Dossier neatly bound in a looseleaf binder or facsimile . . . . .	¼ <input type="checkbox"/>	/0
Cardboard or plastic dividers separate each section . . . . .	¼ <input type="checkbox"/>	/0
Protruding tabs identify the beginning of each section . . . . .	¼ <input type="checkbox"/>	/0
Protruding tabs inserted within Section 'C1' identify the beginning of the source code of each class . . . . .	¼ <input type="checkbox"/>	/0
Double Space all text except source code . . . . .	¼ <input type="checkbox"/>	/0
Footer on every page includes the section and page number. . . . .	¼ <input type="checkbox"/>	/0
Cover Page . . . . .	¼ <input type="checkbox"/>	/0
Refined Table of Contents (Include contents of each section under the title of each section.) . . . . .	¼ <input type="checkbox"/>	/0
Preface: Preamble (Relate the history and personal significance of your experience in developing the dossier.) . . . . .	¼ <input type="checkbox"/>	/0
Preface: Worthiness (Rationale defending dossier program as having a socially redeeming application. Why do it?) . . . . .	¼ <input type="checkbox"/>	/0
Preface: Acknowledgements . . . . .	¼ <input type="checkbox"/>	/0
Preface: Annotated Bibliography (To annotate is to comment, to give your opinion of appropriateness.) . . . . .	¼ <input type="checkbox"/>	/0
Title Page with clip art for each section of the dossier . . . . .	¼ <input type="checkbox"/>	/0
Table of Contents for each section of the dossier . . . . .	¼ <input type="checkbox"/>	/0
Index at the back of the dossier listing the page number(s) of topics by key words . . . . .	¼ <input type="checkbox"/>	/0
No spelling or grammar errors were detected . . . . .	¼ <input type="checkbox"/>	/0
<b>Physical Organization and Presentation . . . . .</b>	<b>/4</b>	<b>/0</b>

## IBO Higher Level Computer Science Mastery Techniques

### All of Which Must Be Non-trivial, Well-documented & Appropriate

1. **Adding data** to an instance of the **RandomAccessFile** class by direct manipulation of the file pointer using the **seek** method. . . . . Page # \_\_\_\_\_
2. **Deleting data** from an instance of the **RandomAccessFile** class by direct manipulation of the file pointer using the **seek** method. (Data primitives or objects may be shuffled or marked as deleted.) . . . . Page # \_\_\_\_\_
3. **Searching for specified data** in an instance of the **RandomAccessFile** class. . . . . Page # \_\_\_\_\_
4. **Recursion.** . . . . . Page # \_\_\_\_\_
5. **Merging two or more sorted data structures.** . . . . . Page # \_\_\_\_\_
6. **Polymorphism.** . . . . . Page # \_\_\_\_\_
7. **Inheritance.** . . . . . Page # \_\_\_\_\_
8. **Encapsulation.** . . . . . Page # \_\_\_\_\_
9. **Parsing a text file or other data stream** . . . . . Page # \_\_\_\_\_
10. Implementing a **hierarchical composite data structure**. A composite data structure in this definition is a class implementing a record style data structure. A hierarchical composite data structure is one that contains more than one element and at least one of the elements is a composite data structure. Examples are, an array or linked list of records, a record that has one field that is another record, or an array. . Page # \_\_\_\_\_
11. **Any five standard level mastery factors:** arrays, user-defined objects, objects as data records, simple selection (if-else), complex selection (nested if, if with multiple conditions or switch), loops, nested loops, user-defined methods, user-defined methods with parameters, user-defined methods with return values, sorting, searching, file I/O, additional libraries, sentinels or flags . . . . . Page # \_\_\_\_\_
- 12-15. Implementation of abstract data types (ADTs) according to "Implementation of ADTs" table:
 

linked list OR tree OR stack OR queue OR array implementation of a hash table. . . . .	Page # _____	{ <input type="checkbox"/>	}
		<input type="checkbox"/>	<input type="checkbox"/>

Mastery Techniques (Minimum of 10 Mastery Techniques Are Required For Full IBO Credit!!) . . . . .	<b><u>/10</u></b>	<b><u>/0</u></b>
Subtotal . . . . .	<b>/14</b>	<b>/0</b>

**Stage A: ANALYSIS**

**A1 Analysing the problem:** A thorough discussion and analysis of the problem which is being solved. .... /8

/4

State the problem to be solved or show some evidence that relevant information has been collected. ....	<input type="checkbox"/>
Describe the problem to be solved.. ....	<input type="checkbox"/>
Provide evidence that information relating to the problem has been collected. ....	<input type="checkbox"/>
Provide evidence that a systematic method has been used in the analysis of the problem. ....	<input type="checkbox"/>
Descriptions of previous attempts to solve the problem by experts in the field:	
1. Analysis of at least one related NONCOMPUTERIZED product or process, past and/or present .....	<input type="checkbox"/>
2. Analyses of at least three related existing computerized products and/or processes .....	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
Transcripts of one interview of potential or actual "clients" or users" of the application .....	<input type="checkbox"/>
UML User Stories .....	<input type="checkbox"/>
UML Use Case Diagram .....	<input type="checkbox"/>
Essay synthesizing the theoretical underpinnings with the above analyses of the application .....	<input type="checkbox"/>

**A2 Criteria for success:** State expected behaviour of the solution and outline the limits under which it can operate  
**Note:** Pay very careful attention to these criteria. Subsequent criteria involve comparing what you set out to do as STATED HERE and what you eventually do: criterion C2 (usability), C4 (success of the program), D2 (evaluating solutions) and D3 (including user documentation). Be realistic in your ambitions!! ..... /8

/4

State some objectives of the solution. ....	<input type="checkbox"/>
Describes most of the objectives of the solution.. ....	<input type="checkbox"/>
Relate all of the objectives of the solution to the analysis of the problem.. ....	<input type="checkbox"/>
Outline the limits under which the solution will operate. ....	<input type="checkbox"/>
1. Tabulate the JUnit boolean assert( ) statements .....	<input type="checkbox"/> <input type="checkbox"/>
2. Time to retrieve data from a file .....	<input type="checkbox"/>
3. Response to invalid data input .....	<input type="checkbox"/>
4. Response to extreme data input .....	<input type="checkbox"/>
5. Quantify the data that can be stored .....	<input type="checkbox"/>
6. Ease of use of input screen .....	<input type="checkbox"/>
7. Appropriate response to user input. ....	<input type="checkbox"/>

Include an initial design . . . . .	<input type="checkbox"/>
Include a prototype which does not correspond to the initial design. . . . .	<input type="checkbox"/>
Include a complete prototype which does corresponds to the initial design . . . . .	<input type="checkbox"/>
Document user feedback in evaluating the prototype . . . . .	<input type="checkbox"/>
UML Class (Structure) Diagram Showing Only Associations . . . . .	<input type="checkbox"/>
UML Class (Structure) Diagram Including Attributes and Operations . . . . .	<input type="checkbox"/>
UML Sequence (Activity) Diagram . . . . .	<input type="checkbox"/>
Functional Prototype: Input/Output . . . . .	<input type="checkbox"/>
Functional Prototype: Implement Direct Access File Handling and/or an Abstract Data Type . . . . .	<input type="checkbox"/>
Document user comments on reaction to the prototype . . . . .	<input type="checkbox"/>

**Stage B: DETAILED DESIGN**

**B1 Data structures:** Support the data-storage requirements. Allow clear, efficient algorithms. Data-logical classes and data appropriate methods. Include class definitions, file structures, abstract data types and consideration of alternatives.

Outline some of the data structures/types to be used in the solution. (cf A2) . . . . .	<input type="checkbox"/>
Describe some of the data structures/types to be used and provide sample data (cf A2).. . . . .	<input type="checkbox"/>
Discuss ALL of the data structures/types to be used and provide sample data (cf A2). . . . .	<input type="checkbox"/>
Clearly ILLUSTRATE all of the data structures/types to be used and provide sample data for all of them (cf A2) . . . . .	<input type="checkbox"/>
Illustrate AND annotate the data structure record(s) and their fields to be used in the program . . . . .	<input type="checkbox"/>
CHOOSE ONE OF a linked list OR tree OR stack OR queue OR array implementation of a hash table. Then:	
▶ Defend the use of a linked list. (Provide examples of source code to be used.) OR . . . . .	<input type="checkbox"/>
▶ Defend the use of a binary search tree. (Provide examples of source code to be used.) OR . . . . .	<input type="checkbox"/>
▶ Defend the use of a stack. (Provide examples of source code to be used.) OR . . . . .	<input type="checkbox"/>
▶ Defend the use of a queue. (Provide examples of source code to be used.) OR . . . . .	<input type="checkbox"/>
▶ Defend use of an array implementation of a hash table. (Provide examples of source code to be used.) . . . . .	<input type="checkbox"/>
Provide Sketches/illustrations of the above Abstract Data Type(s) to be used in the program . . . . .	<input type="checkbox"/>
Defend the use of Direct Access File Handling (Provide examples of source code to be used.) . . . . .	<input type="checkbox"/>
Provide Sketches/illustrations of all using Direct Access File Handling in the program . . . . .	<input type="checkbox"/>

**B2 Algorithms:** Choose algorithms, classes and methods that support the objectives (A2) and required data structures (B1). Include parameters, return values, and descriptions of pre- and post-conditions .....

/4

/4

Outline some of the algorithms to be used in the solution. (cf A2) .....	<input type="checkbox"/>
Describe most of the algorithms to be used with details of parameters and return values.(cf A2).. .....	<input type="checkbox"/>
Discuss ALL algorithms to be used with parameters, return values, pre-conditions and post-conditions.(cf A2). ...	<input type="checkbox"/>
Algorithms are sufficiently logical, detailed, and well documented to be used to create the solution in Java. ....	<input type="checkbox"/>
Javadoc statements of purpose for all classes, constructors and methods .....	½ <input type="checkbox"/>
Signatures of all constructors contain their formal parameters and corresponding data types .....	½ <input type="checkbox"/>
Signatures of all methods contain their formal parameters and corresponding data types .....	½ <input type="checkbox"/>
Javadoc @author tags specify all authors and sources of all classes .....	½ <input type="checkbox"/>
Javadoc @pre tags explain preconditions of methods. ....	½ <input type="checkbox"/>
Javadoc @post tags explain post-conditions of methods .....	½ <input type="checkbox"/>
Javadoc @param tags explain all significant formal parameters in the signatures. ....	½ <input type="checkbox"/>
Javadoc @return tags specify all values returned by all significant methods .....	½ <input type="checkbox"/>

**B3 Modular organization:** Incorporate the data structures and methods (criteria B1 and B2) in a logical way Show connections between modules (hierarchical decomposition or class dependencies), algorithms and data structures. ...

/6

/4

Outline some of the modules to be used in the solution. ....	<input type="checkbox"/>
Describe most of the modules to be used, showing connections between them.. .....	<input type="checkbox"/>
Describe all of the modules to be used and show the connections to data structures and methods. ....	<input type="checkbox"/>
Modules are sufficiently logical, detailed and well documented to be used to create the solution in Java.. ....	<input type="checkbox"/>
System Flowchart (Know and use standard flowchart symbols.) .....	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
1. Stepwise Refinement in System Design Specifications or CRC Cards .....	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>

**Stage C: The program**

**C1 Using good programming style:** Program listings are easily readable, include small and clearly structured Java methods, sufficient and appropriate comments, meaningful identifier names and a consistent indentation scheme. **[Source Code]** /3 /3

The program listing demonstrates <b>some attention</b> to good programming style. ....	<input type="checkbox"/>
The program listing <b>mostly demonstrates</b> attention to good programming style.. ....	<input type="checkbox"/>
<b>All parts</b> of the program listing demonstrate considerable attention to good programming style. ....	<input type="checkbox"/>
The beginning of every file of source code contains an identification header .....	¼ <input type="checkbox"/>
Every constant, type and variable declaration has an explanatory comment. ....	¼ <input type="checkbox"/>
All identifier names must be "meaningful" .....	¼ <input type="checkbox"/>
All objects are clearly separated and have comments for their value parameters .....	¼ <input type="checkbox"/>
Suitable indentation illustrates and delineates the various programming constructs .....	¼ <input type="checkbox"/>
Sun MicroSystems' convention for naming a package of classes is adhered to in all cases .....	¼ <input type="checkbox"/>
Use syntax (black or colour) highlighting .....	¼ <input type="checkbox"/>
Number all lines of source code .....	¼ <input type="checkbox"/>
Header and/or Footer contains the program name, page number, and date of last revision. ....	¼ <input type="checkbox"/>
Wrapping of lines is done with offsetting indentation .....	¼ <input type="checkbox"/>
All methods are small and clearly structured .....	¼ <input type="checkbox"/>
All classes declare all instance variables to be private and most or all methods to be public .....	¼ <input type="checkbox"/>

**C2 Usability:** Include features that make the program more user-friendly, such as helpful menus, help instructions, useful guidance to the user during the execution of the program. This criterion does not refer to internal error checking. ... /7 /3

Document some user-friendly features within the program. ....	<input type="checkbox"/>
Document fully the user-friendly features within the program. ....	<input type="checkbox"/>
The program meets the usability objectives in criterion A2. ....	<input type="checkbox"/>
Input and output is addressed with a Graphical User Interface using classes such as AWT &/or Swing .....	<input type="checkbox"/>
Help (directions, explanations) are retrievable with GUI constructed components .....	<input type="checkbox"/>
Provide a meaningful tool tip as appropriate for every JComponent : obj.setToolTipText("Click"); .....	<input type="checkbox"/>
Relate each usability feature with the relevant objectives presented in criterion A2:	
▶ Reproduce relevant parts from the monitor that generate the usability .....	<input type="checkbox"/>
▶ Annotate relevant parts from the monitor that generate the menus and instructions. ....	<input type="checkbox"/>
▶ Reproduce relevant parts of the source code that generate the menus and instructions .....	<input type="checkbox"/>
▶ Annotate relevant parts of the source code that generate the menus and instructions .....	<input type="checkbox"/>

**C3 Handling errors:** Detect and reject erroneous data input from the user and prevent common runtime errors caused by calculations and data-file errors. ....

/10

/3

Document a few error-handling facilities in the program or document only one type of input or output.. ....	<input type="checkbox"/>
Document many error-handling facilities in the program and document more than one type of input or output... ..	<input type="checkbox"/>
Document the error-handling of each input and output method within the program. ....	<input type="checkbox"/>
Using Exception Handling (performance in try blocks, response in catch or finally blocks), do:	
▶ Provide annotated screen shots from the monitor that illustrates the handling of Runtime Exceptions .....	<input type="checkbox"/>
▶ Provide corresponding annotated source code that illustrates the handling of Runtime Exceptions .....	<input type="checkbox"/>
▶ Provide annotated screen shots from the monitor that illustrates the handling of Arithmetic Exceptions. ....	<input type="checkbox"/>
▶ Provide corresponding annotated source code that illustrates the handling of Arithmetic Exceptions. ....	<input type="checkbox"/>
▶ Provide annotated screen shots from the monitor that illustrates the handling of an IOException showing how the program recovers if the user enters data different from that expected .....	<input type="checkbox"/>
▶ Provide corresponding annotated source code that illustrates the handling of an IOException showing how the program recovers if the user enters data different from that expected .....	<input type="checkbox"/>
▶ Provide annotated screen shots from the monitor that illustrates the handling of an IOException showing how the program recovers if it tries to access a file that does not exist .....	<input type="checkbox"/>
▶ Provide corresponding annotated source code that illustrates the handling of an IOException showing how the program recovers if it tries to access a file that does not exist .....	<input type="checkbox"/>
▶ Provide annotated screen shots from the monitor that illustrates an extension of the Exception class by handling an error that is particular to the program and not a standard API exception .....	<input type="checkbox"/>
▶ Provide corresponding annotated source code that illustrates an extension of the Exception class by handling an error that is particular to the program and not a standard API exception .....	<input type="checkbox"/>

**C4 Success of the program:** Incorporate the data structures and methods (criteria B1 and B2) in a logical way Show connections between modules (hierarchical decomposition or class dependencies), algorithms and data structures. This is to be demonstrated by **annotated** screen shots of sample runs of the program .....

/6

/3

The student successfully achieved <b>SOME</b> of the objectives from criterion A2 .....	<input type="checkbox"/>
The student successfully achieved <b>MOST</b> of the objectives from criterion A2 .....	<input type="checkbox"/>
The student successfully achieved <b>ALL</b> of the objectives from criterion A2 .....	<input type="checkbox"/>
SOME objectives from criterion A2 are documented as achieved by screen shots of sample runs of the program. .	<input type="checkbox"/>
MOST objectives from criterion A2 are documented as achieved by screen shots of sample runs of the program ..	<input type="checkbox"/>
ALL objectives from criterion A2 are documented as achieved by screen shots of sample runs of the program ....	<input type="checkbox"/>
SOME objectives from criterion A2 are achieved when the student ran the program for the teacher. ....	<input type="checkbox"/>
MOST objectives from criterion A2 are achieved when the student ran the program for the teacher. ....	<input type="checkbox"/>
ALL objectives from criterion A2 are achieved when the student ran the program for the teacher. ....	<input type="checkbox"/>

**Stage D: Documentation**

**D1 Include an annotated hard copy of the test output:** Hard copy of test output demonstrates that the program fulfills the criteria for success in criterion A2. The output must be annotated. Testing must be done as claimed in the documentation.

/5

/4

Include an incomplete set of sample output. . . . .	<input type="checkbox"/>
Include an incomplete set of annotated sample output. . . . .	<input type="checkbox"/>
Include a mostly complete set of annotated sample output. . . . .	<input type="checkbox"/>
Include a complete set of annotated sample output, testing all the objectives in criterion A2 . . . . .	<input type="checkbox"/>
Organize GUI screen shots and annotations in 2 side-by-side column tables . . . . .	<input type="checkbox"/>
Sample output tests ALL the criteria for success that were stated in section A2 . . . . .	<input type="checkbox"/>
Includes responses to VALID input “within the limits under which the solution will operate” . . . . .	<input type="checkbox"/>
Includes responses to INVALID input “within the limits under which the solution will operate” . . . . .	<input type="checkbox"/>
Sample output includes all the different branches of the program’s possible executions . . . . .	<input type="checkbox"/>

**D2 Evaluating solutions:** Hard copy of test output demonstrates that the program fulfills the criteria for success in criterion A2. The output must be annotated. Testing must be done as claimed in the documentation. . . . .

/6

/4

Only <b>outline</b> the solution. . . . .	<input type="checkbox"/>
<b>Outline</b> the solution and <b>partially</b> consider effectiveness, efficiency and possible improvements.. . . .	<input type="checkbox"/>
<b>Discuss</b> the effectiveness and efficiency of the solution and <b>suggest</b> alternative processes and improvements. . . .	<input type="checkbox"/>
Suggest alternative approaches to the solution and the design process. . . . .	<input type="checkbox"/>
Effectiveness is discussed in relation to the original description of the problem . . . . .	<input type="checkbox"/>
Efficiency is discussed in general terms without reference to Big-O measures. Discuss garbage collection . . . . .	<input type="checkbox"/>
Indicate the Big-O efficiency measures of algorithms and the circumstances where Big-O measures may vary . . . .	<input type="checkbox"/>
Distinguish machine efficiency from programmer efficiency. . . . .	<input type="checkbox"/>
Discussion of improvements and extensions for future development is realistic . . . . .	<input type="checkbox"/>
Discuss alternative approaches to addressing the problem . . . . .	<input type="checkbox"/>

**D3 Include user documentation:** Include sample output, written instructions, and screen shots to illustrate program operation that are separate from the hard copy in criterion D1. The user documentation must be hard copy. . . . . /12

/3

Include some user documentation. . . . .	<input type="checkbox"/>
Include user documentation that contains clear instructions about running the program.. . . .	<input type="checkbox"/>
Include user documentation that contains clear, illustrated instructions about installing and running the program. . . . .	<input type="checkbox"/>
Cover Page . . . . .	½ <input type="checkbox"/>
Table of Contents (Not of your dossier, but of this User's Manual.) . . . . .	½ <input type="checkbox"/>
Footer on every page with Name of Manual, Current Page Number and Total Number of Pages. . . . .	½ <input type="checkbox"/>
System Requirements (Must Java 2 or Sun's Applet Browser be installed on the system?) . . . . .	½ <input type="checkbox"/>
Installation (Start with "Insert the CD." Assume the user just discovered the existence of a computer.) . . . . .	½ <input type="checkbox"/>
Overview (Describe what the application does.) . . . . .	½ <input type="checkbox"/>
Features . . . . .	½ <input type="checkbox"/>
Options (choices: click a checkbox, type a word, scroll, etc.) . . . . .	½ <input type="checkbox"/>
Using Help . . . . .	½ <input type="checkbox"/>
How to start up . . . . .	½ <input type="checkbox"/>
How to view (browse, scroll through) existing records . . . . .	½ <input type="checkbox"/>
How to search for an existing record . . . . .	½ <input type="checkbox"/>
How to create a record and/or database . . . . .	½ <input type="checkbox"/>
How to delete a record and/or database . . . . .	½ <input type="checkbox"/>
How to save a record and/or database . . . . .	½ <input type="checkbox"/>
How to retrieve a record and/or database . . . . .	½ <input type="checkbox"/>
How to change (update) a record and/or database . . . . .	½ <input type="checkbox"/>
How to quit (shut down) the program properly . . . . .	½ <input type="checkbox"/>
Technical Information . . . . .	½ <input type="checkbox"/>
How and from where to get updated versions . . . . .	½ <input type="checkbox"/>
FAQ (Frequently Asked Questions) . . . . .	½ <input type="checkbox"/>
Support in terms of email address, snail address, phone number, fax number, ICQ#, website . . . . .	½ <input type="checkbox"/>
Index (Not of your dossier, but of this User's Manual.) . . . . .	½ <input type="checkbox"/>
Clip art illustrating the user documentation . . . . .	½ <input type="checkbox"/>



**Stage E: Holistic approach**

**E Holistic approach to the dossier:** The dossier is an ongoing process involving consultation between the student and teacher. The student is aware of the expectations of the teacher from the beginning of the process and each achievement level awarded is justified by a written comment from the teacher at the time of marking. The examples given below for each criterion level are teacher orientated and each teacher should use discretion when judging the levels.. . . . . /13

/3

The student showed minimal commitment, participating minimally in class discussions, made most deadlines, had some discussion initiated by the teacher but did not exploit opportunities to develop or improve the dossier . . . . .	<input type="checkbox"/>
The student showed good commitment, participating in class discussions, initiating discussions with the teacher and/or the rest of the class and/or became fully involved in the development of the dossier. . . . .	<input type="checkbox"/>
The student showed full commitment, participating fully in class discussions, taking initiatives both in discussion with the teacher and/or the rest of the class and in subsequent work of a more independent nature and/or demonstrated a full understanding of all the steps in the development of his/her dossier . . . . .	<input type="checkbox"/>
Focus: Student took every reasonable opportunity to advance the progress and quality of the dossier . . . . .	<input type="checkbox"/>
Attitude: Student remained optimistic that his/her dossier would be done well and on time . . . . .	<input type="checkbox"/>
Attitude: Student encouraged others to remain optimistic that their dossiers would be done well and on time . . . . .	<input type="checkbox"/>
Respect: Student showed tact and continued respect for the worthiness of the dossier experience . . . . .	<input type="checkbox"/>
Respect: Student offered constructive criticism for advancing the worthiness of the dossier experience . . . . .	<input type="checkbox"/>
Participation: Student worked well with the rest of the class . . . . .	<input type="checkbox"/>
Participation: Student made helpful suggestions to other students regarding their dossiers . . . . .	<input type="checkbox"/>
Understanding: Student demonstrated a full understanding of the concepts associated with his/her dossier . . . . .	<input type="checkbox"/>
Initiative: Student often approached the teacher to discuss dossier issues . . . . .	<input type="checkbox"/>
Initiative: Student was always willing to try a different approach . . . . .	<input type="checkbox"/>
Perseverance: Student thought a lot about the problems encountered before seeking assistance. . . . .	<input type="checkbox"/>
Insight: Student showed an accurate and intuitive understanding of how to proceed in developing the dossier . . . . .	<input type="checkbox"/>
Deadlines: Student met <b>teacher set</b> deadlines by setting and meeting earlier <b>student set</b> target dates . . . . .	<input type="checkbox"/>

**Total Possible Marks From Assessment Evaluation *Before* Applying The Mastery Factor . . . . .     /100         /50**

# THE MASTERY FACTOR

The **assigned dossier mark** out of 50 is calculated by multiplying the **Mastery Factor** by the **IB Assessment Evaluation**. Of fifteen programming aspects, the mastery factor is a direct reflection of the number of those techniques appropriately applied in the dossier.

Number of Aspects in Which the Student Demonstrates Mastery	Mastery Factor
10 or more	1
9	0.9
8	0.8
7	0.7
6	0.6
5	0.5
4	0.4
3	0.3
0, 1 or 2	0.2

Assigned IBO Dossier Mark = Mastery Factor [      ] X Assessment Evaluation [      ] =                      /50

## COLLABORATIVE WORK IS FORBIDDEN.

A teacher is expected to give “educational guidance”, but the work of the dossier must be “that of the candidate alone.”

However, The Use and Acknowledgment of Code by Another Is Encouraged and A Sign Of Strong Scholarship!

IBO Dossier Java Programs necessarily use code that the Candidates themselves did not “create”. Such code must be either “publically” recognizable as standard by a professional practitioner, as with Java’s API classes, or credit must be given with both internal comments and reference in the dossier’s Acknowledgements section.

### How Can You Learn From Other Students or Teachers When You Are Stuck on Your Dossier and Do Not Know What to Do?

Define the problem that you are having difficulty with as clearly and completely as possible. Restate the generics and mechanics of that problem in the context or scenario of an entirely different problem that is unrelated to your dossier program. You may then ask a teacher or another student to help you solve that unrelated problem. You should then study the solution to that different problem as a means to understanding the techniques or structures or skills that solved that problem. You may then apply your newly acquired knowledge of techniques or structures or skills to your dossier without a need for collaboration.

For example, if while doing your dossier, you realize that you can’t properly delete a node in your linked list, then define your problem as learning a technique to appropriately delete a node in a linked list. You may then ask your teacher for references in a textbook where you may read about deleting nodes from linked lists, and you may ask your teacher for an exercise in a text book that requires the deletion of a node from a linked list. You may then try to solve the textbook exercise yourself or you may elicit collaboration from your teacher or a fellow student in solving the textbook exercise. Whatever you do, focus on understanding the technique sufficiently so that you may apply your new understanding to your dossier without collaboration.

This is a valid part of the IB learning process because it is recognized that everyone, either earlier or later, must learn the mastery “aspects” required by the dossier from an established source. Nonexistent are students who somehow, willy-nilly, created a language equivalent to Java all by themselves or who even “intuitively” knew how to implement direct access files or linked lists or binary search trees or stacks or queues or an array implementation of a hash table. We all stand on the shoulders of giants.