

Table of Contents

Example of IB Computer Science Dossier Table of Contents, March 14, 2004, Gerry Donaldson, Calgary, Canada

Preface	P-1
Preamble	P-1
Acknowledgements	P-4
Bibliography of Material That Influenced This Dossier	P-5
Mastery Techniques Specified	MT-1
1. Insert a new data item into a Linked List	MT-2
2. Delete a data item from a Linked List	MT-3
3. Search for a data item in a Linked List	MT-4
4. Add directly a new record to a File	MT-5
5. Delete directly a record from a File	MT-6
6. Search directly for a record in a File	MT-7
7. Merge two sorted Data Structures	MT-8
8. User-defined functions	MT-9
9. Using pointers as Parameters	MT-10
10. Recursion with a Binary Search	MT-11
11. Using more than one kind of composite data structure: classes and structs .	MT-12
Criterion A: Analyzing the Problem	A-1
1. Review: Manual Swiss Pairings At Sir Winston Churchill High School	A-2
2. Review: Swiss Perfect 98 Build 408	A-5
3. Review: MacMahon	A-7
4. Review: Swiss Manager	A-9
Criterion B: Documenting the Design Process	B-1
Stepwise Refinement in System Design Specifications:	
1. System Objectives.	B-2
2. System Context.	B-3
3. Functional Requirements, which include the performance subsystem.	B-4
4. Quality Requirements.	B-6
5. Future Requirements.	B-7
System Flowchart (Know and use standard flowchart symbols.)	B-8
UML Diagrams:	
1. Class Diagram showing the detailed attributes and operations.	B-9
2. Class Diagram without detailed attributes.	B-10
3. Collaboration Diagrams	B-11
4. Sequence Diagrams.	B-12
5. Statechart Diagrams.	B-13
6. Use Cases (textual) of Different Scenarios: happy line and alternate flows. .	B-14
7. Use Case Context Diagram(s): summary of use case stories.	B-16
Pseudo-Code of non-standard and non-trivial modules.	B-17

Criterion C: Using Appropriate Data Structures	C-1
1. Linked List preferred for adding and deleting records	C-1
2. Direct File Handling preferred rapid access of secondary storage	C-2
3. Classes and Objects Improve Modularization	C-3
Criterion D: Using Efficient Algorithms	D-1
Criterion E: Designing a Testing Strategy	E-1
Criterion F: Annotated Hard Copy of the Test Output	F-1
Criterion G: User-Friendly Features	G-1
Criterion H: Handling Errors	H-1
Criterion I: Implementing the Program	I-1
Criterion J: Using Good Programming Style (<i>Source Code Listing</i>)	J-1
1. swissDrv.cpp (main line)	J-2
2. swissPrg.h (documentation of swiss source code functions)	J-3
3. swissPrg.cpp (implementation of swiss source code functions)	J-4
4. swisslst.h (definition and documentation of swiss linked list class)	J-6
5. swisslst.cpp (implementation of swiss linked list class)	J-8
6. Swissfil.h(definition and documentation of swiss direct file handling class) ..	J-10
7. swisslst.cpp (implementation of swiss direct file handling class)	J-11
8. reg.dat (sample data file compatible with the registration subsystem)	J-14
9. pair.dat (sample data file compatible with the swiss pairing subsystem)	J-15
Criterion K: Evaluating Solutions	K-1
1. Registration Is Rapid and Thorough	K-2
2. Pairings Are Rapid and Accurate	K-3
3. Reports Are Clear and Flexible	K-4
4. Future Enhancements	K-5
5. JavaDoc Documentation	K-6
Criterion L: User Documentation	L-1
1. Minimal System Requirements	L-2
2. Obtaining Support	L-2
3. Getting Started	L-3
4. Registering Participants	L-4
5. Pairing the Rounds	L-6
6. Reporting Results	L-8
7. Printing Tables	L-9
Appendix: Rules of Formal Debate, Alberta Speech and Debate Association	A-1