## PROJECT CODING RUBRIC SCORE BREAK DOWN:

Compilable   1   Compiles with no errors	Checklist	<b>Points Earned</b>	Where in Code it is found
Runtime Errors    1		1	
Readability		1	
Math Technical Knowledge from other areas   2	Readability	1	
see how much increase/decrease was earned		2	Use of counters to keep count of bets, multiplication to
Independency	_		calculate payouts, percentage difference (economics) to
Code Organization   1			see how much increase/decrease was earned
Robustness Resource Usages Resource Int (usigned) Resour		1	Compilable on any OSes using any IDEs or compilers
Robustness   1   Code captures edge-cases and wrong inputs	Code Organization	1	
No memory leaks/reasonable RAM and CPU consumption   Write, Compile, and Include Your Own Libraries			
Write, Compile, and Include Your Own Libraries   Did not compile a personal library (used an external 3rd party library OpenSDL)		1	
Libraries  All Primitive Data Types  0.5  Example where each is used:  • Char • Int (unsigned) • Float • Double • String  Global and Static Variables  0.5  Global Variables (in main.cpp): • bool imageDisplayed – line 5 • std::string projectName – line6 • int initialBalance – line7  Static Variables  0.5  Used bitwise XOR operator (*^*) • Found in randomNumber.cpp in line 5  Conditional AND Iterative Statements  0.5  Use of Conditional (IF) and Iterative (WHILE) statements can both be found in main.cpp  Functions  0.5  Total of 9 functions  BettingOptions() • BettingOptionParameters() • Image() • initializePlayerData() • main() • output_DealerName() • randomNumber() • writePlayerDataToFile()  Preprocessor  1  Used all of the following preprocessors: • #include <cstdib> • #include <cstdib> • #include <cstdib> • #include <ctime> • #include <firm> •</firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></firm></ctime></cstdib></cstdib></cstdib>			
Char   Int (unsigned)   Int (Signed)   Int (Signe		0	
Int (unsigned)   Int (Signed)   Int (Signed)   Float   Double   String   Global and Statie Variables   O.5   Global Variables (in main.cpp):   bool imageDisplayed – line 5   std::string projectName – line6   int initialBalance – line7   Static Variables   Static Variables   Static int dealer last name – line 41   Use distivus XOR operator ('^')   Found in randomNumber.cpp in line 5   Conditional AND Iterative Statements   O.5   Use of Conditional (IF) and Iterative (WHILE) statements can both be found in main.cpp  Functions   O.5   Total of 9 functions   BettingOptions()   EndOrGameExpression()   getBettingOptionParameters()   Image()   initializePlayerData()   main()   output_DealerName()   randomNumber()   randomNumber()   writePlayerDataToFile()   Used all of the following preprocessors:   #include <cstraips #<="" #include="" <cstrains="" <cstraips="" td=""  =""><td>All Primitive Data Types</td><td>0.5</td><td></td></cstraips>	All Primitive Data Types	0.5	
Int (Signed)   Float     Double     Double     String     Global and Static Variables     Global Variables     Global Variables     Global Variables     Global Variables     String     Static Variables     Static Vari			• Char
Global and Static Variables  Global String  Global Variables (in main.cpp):			• Int (unsigned)
Global and Static Variables  Global String Global Variables (in main.cpp):			• Int (Signed)
Global and Static Variables  0.5  Global Variables (in main.cpp):  bool imageDisplayed – line 5  std::string projectName – line6  int initialBalance – line7  Static Variables  0.5  Used bitwise XOR operator ('^')  Found in randomNumber.pp in line 5  Conditional AND Iterative Statements  0.5  Use of Conditional (IF) and Iterative (WHILE) statements can both be found in main.cpp  Functions  0.5  Total of 9 functions  BettingOptions()  EndOfGameExpression()  getBettingOptionParameters()  Image()  initializePlayerData()  main()  output_DealerName()  randomNumber()  writePlayerDataToTie()  Preprocessor  1  Used all of the following preprocessors:  #include <istring> #include <istring> #include <istring> #include <istring> #include <ifiteam> #include</ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></ifiteam></istring></istring></istring></istring>			• Float
Global and Static Variables    Global Variables   Side   S			Double
• bool imageDisplayed – line 5 • std::string projectName – line6 • int initialBalance – line7 State Variables • static int dealer last name – line 41  Bitwise Operators  0.5  Used bitwise XOR operator (**) • Found in randomNumber.cpp in line 5  Conditional AND Iterative Statements  0.5  Use of Conditional (IF) and Iterative (WHILE) statements can both be found in main.cpp  Functions  1 total of 9 functions • BettingOptions() • EndOfGameExpression() • getBettingOptionParameters() • Image() • initializePlayerData() • main() • output_DealerName() • main() • writePlayerDataToFile()  Preprocessor  1 Used all of the following preprocessors: • #include < ctime> • #include < ctime> • #include < ttimp> • #include < fstream> • #include < farray> • #include < thread>			
Std::string projectName - line6   int initialBalance - line7	Global and Static Variables	0.5	
Bitwise Operators  0.5  Used bitwise XOR operator (*^*)  Found in randomNumber.cpp in line 5  Use of Conditional (IF) and Iterative (WHILE) statements can both be found in main.cpp  Functions  0.5  Total of 9 functions  BettingOptions()  EndOfGameExpression()  getBettingOptionParameters()  Image()  initializePlayerData()  main()  output_DealerName()  randomNumber()  writePlayerDataToFile()  Preprocessor  1  Used all of the following preprocessors:  #include <ctime>  #include <ctime>  #include <ctime>  #include <firting>  #include <firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></firting></ctime></ctime></ctime>			
Bitwise Operators  0.5 Used bitwise XOR operator (*^*)  • Found in randomNumber.cpp in line 5  Conditional AND Iterative Statements  0.5 Use of Conditional (IF) and Iterative (WHILE) statements can both be found in main.cpp  Functions  0.5 Total of 9 functions  • BettingOptions() • EndOfGameExpression() • getBettingOptionParameters() • Image() • initializePlayerData() • main() • output_DealerName() • randomNumber() • writePlayerDataToFile()  Preprocessor  1 Used all of the following preprocessors: • #include <strices #<="" #include="" <strices="" <strings="" td="" •=""><td></td><td></td><td>• std::string projectName – line6</td></strices>			• std::string projectName – line6
Bitwise Operators  0.5  Used bitwise XOR operator (*^*)  Found in randomNumber.cpp in line 5  Use of Conditional (IF) and Iterative (WHILE) statements can both be found in main.cpp  Functions  0.5  Total of 9 functions  BettingOptions()  EndOfGameExpression()  getBettingOptionParameters()  Image()  initializePlayerData()  main()  output_DealerName()  randomNumber()  writePlayerDataToFile()  Preprocessor  1  Used all of the following preprocessors:  #include <istream> #include <itime> #include &lt;</itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></itime></istream>			• int initialBalance – line7
Bitwise Operators  O.5  Used bitwise XOR operator ('^') Found in randomNumber.cpp in line 5  Use of Conditional (IF) and Iterative (WHILE) statements can both be found in main.cpp  Functions  O.5  Total of 9 functions  BettingOptions() EndOfGameExpression() getBettingOptionParameters() Image() initializePlayerData() main() output_DealerName() randomNumber() randomNumber() wirePlayerDataToFile()  Preprocessor  1  Used all of the following preprocessors: #include <istream> #include <string> #include <tipre> #include <ti>#include <tipre> #include <tipre> #include <ti>#include <tipre< td=""><td></td><td></td><td></td></tipre<></ti></tipre></tipre></ti></tipre></tipre></tipre></ti></tipre></tipre></tipre></ti></tipre></tipre></tipre></ti></tipre></tipre></tipre></ti></tipre></tipre></tipre></ti></tipre></tipre></tipre></ti></tipre></tipre></tipre></ti></tipre></tipre></tipre></ti></tipre></tipre></tipre></ti></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></tipre></string></istream>			
Conditional AND Iterative Statements  0.5 Use of Conditional (IF) and Iterative (WHILE) statements can both be found in main.cpp  Functions  0.5 Total of 9 functions  BettingOptions()  EndOfGameExpression()  EndOfGameExpression()  Image()  initializePlayerData()  main()  output_DealerName()  randomNumber()  vritePlayerDataToFile()  Preprocessor  1 Used all of the following preprocessors:  #include <citdlib> #include <ctidlib> #include <ctidlib< tid=""> #include <ctidlib> #include <ctidlib< td=""> #include <ctidlib< td=""> #include <ctidlib> #include <ctidlib< td=""> #include <ctidlib< tid=""> #include <ctidlib< td=""> #include <ctidlib< td=""> #include <ctidlib< td=""></ctidlib<></ctidlib<></ctidlib<></ctidlib<></ctidlib<></ctidlib<></ctidlib<></ctidlib<></ctidlib<></ctidlib<></ctidlib<></ctidlib></ctidlib<></ctidlib<></ctidlib></ctidlib<></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></ctidlib></citdlib>			
Conditional AND Iterative Statements  0.5  Use of Conditional (IF) and Iterative (WHILE) statements can both be found in main.cpp  Total of 9 functions  BettingOptions()  EndOfGameExpression()  getBettingOptionParameters()  Image()  initializePlayerData()  main()  output_DealerName()  randomNumber()  writePlayerDataToFile()  Preprocessor  1  Used all of the following preprocessors:  #include <iostream> #include <ctime> #include <ctime> #include <string> #include <tweto> #include <tweto> #include <tweto> #include <tweto> #include <tuple> #include <tri>#include <try> #include <try #include="" <tr="" <try=""> #include  #inc</try></try></tri></tuple></tweto></tweto></tweto></tweto></string></ctime></ctime></iostream>	Bitwise Operators	0.5	
Functions  0.5  Total of 9 functions  BettingOptions() EndOfGameExpression() EndOfGameExpression() Image() Image() Image() ImitializePlayerData() Image() ImitializePlayerData() Image() ImitializePlayerData() Image() ImitializePlayerData() Image()			
Functions  0.5  Total of 9 functions  BettingOptions() EndOfGameExpression() getBettingOptionParameters() Image() initializePlayerData() main() output_DealerName() randomNumber() writePlayerDataToFile()  Preprocessor  1  Used all of the following preprocessors: #include <iostream> #include <stdiib> #include <ctime> #include <string> #include <string> #include <li>#include <li>#include <li>#include <iuple> #include <fstream> #in</fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></iuple></li></li></li></string></string></ctime></stdiib></iostream>	Conditional AND Iterative Statements	0.5	
Preprocessor  I Used all of the following preprocessors:  #include <stdlib> #include <string> #include <string> #include <string> #include <string> #include <string> #include <string> #include <fistream> #include <string> #include <string> #include <string> #include <fistream> #include <fistream> #include <string> #include <fistream> #include <array> #incl</array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></fistream></fistream></fistream></fistream></fistream></fistream></fistream></fistream></fistream></fistream></fistream></fistream></fistream></fistream></string></fistream></fistream></string></string></string></fistream></string></string></string></string></string></string></stdlib>	Functions	0.5	
Preprocessor  1  Used all of the following preprocessors:  #include <ctime> #include <ctime> #include <ctime> #include <string> #include <li>#include <fstream> #include <string> #include <li>#include <may #include="" <string=""> #include <li>#include <may #include="" <string=""> #include <string> #include <stream> #include <stream> #include <string> #include <stri< td=""><td>1 diletions</td><td>0.5</td><td></td></stri<></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></string></stream></stream></string></may></li></may></li></string></fstream></li></string></ctime></ctime></ctime>	1 diletions	0.5	
• getBettingOptionParameters() • Image() • initializePlayerData() • main() • output_DealerName() • randomNumber() • writePlayerDataToFile()  Preprocessor  1 Used all of the following preprocessors: • #include <iostream> • #include <ctime> • #include <ctime> • #include <string> • #include <string> • #include <tiple> • #include <fstream> • #include <fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></fstream></tiple></string></string></ctime></ctime></iostream>			
Preprocessor  1  Used all of the following preprocessors:  #include <ctime> #include <string> #include <string> #include <string> #include <firms> #include <fi< td=""><td></td><td></td><td></td></fi<></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></firms></string></string></string></ctime>			
• initializePlayerData() • main() • output_DealerName() • randomNumber() • writePlayerDataToFile()  Preprocessor  1 Used all of the following preprocessors: • #include <iostream> • #include <ctime> • #include <string> • #include <li>#include <li>#include <li>#include <li>#include <string> • #include <tuple> • #include <fstream> • #include <array> • #include <array< td=""><td rowspan="6"></td><td></td><td></td></array<></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></array></fstream></fstream></fstream></fstream></fstream></tuple></string></li></li></li></li></string></ctime></iostream>			
<pre>main() output_DealerName() randomNumber() writePlayerDataToFile()  Preprocessor  1</pre>			
<ul> <li>output_DealerName()</li> <li>randomNumber()</li> <li>writePlayerDataToFile()</li> </ul> Preprocessor <ul> <li>Used all of the following preprocessors:</li> <li>#include <iostream></iostream></li> <li>#include <ctime></ctime></li> <li>#include <vector></vector></li> <li>#include <string></string></li> <li>#include <li>string&gt;</li> <li>#include <li>limits&gt;</li> <li>#include <tuple></tuple></li> <li>#include <fstream></fstream></li> <li>#include <fstream></fstream></li> <li>#include <array></array></li> <li>#include <array></array></li> <li>#include <thread></thread></li> </li></li></ul>			
• randomNumber() • writePlayerDataToFile()  Preprocessor  1 Used all of the following preprocessors: • #include <iostream> • #include <ctime> • #include <string> • #include <li>#include <limits> • #include <tuple> • #include <tuple> • #include <fstream> • #include <fstream> • #include <array> • #include <array> • #include <thread></thread></array></array></fstream></fstream></tuple></tuple></limits></li></string></ctime></iostream>			
• writePlayerDataToFile()  Preprocessor  1 Used all of the following preprocessors:  • #include <iostream>  • #include <ctime>  • #include <string>  • #include <li>#include <li>#include <li>* #include <string>  • #include <string>  • #include <fstream>  • #include <tuple>  • #include <fstream>  • #include <fstream>  • #include <array>  • #include <array>  • #include <thread></thread></array></array></fstream></fstream></tuple></fstream></string></string></li></li></li></string></ctime></iostream>			
Preprocessor  Used all of the following preprocessors:  #include <iostream> #include <ctime> #include <string> #include <li>#include <li>#include <li>#include <string> #include <tuple> #include <fstream> #include <fstream> #include <fstream> #include <fragex> #include <array> #include <thread></thread></array></fragex></fstream></fstream></fstream></tuple></string></li></li></li></string></ctime></iostream>			
<ul> <li>#include <iostream></iostream></li> <li>#include <ctime></ctime></li> <li>#include <vector></vector></li> <li>#include <string></string></li> <li>#include <li>limits&gt;</li> <li>#include <tuple></tuple></li> <li>#include <fstream></fstream></li> <li>#include <regex></regex></li> <li>#include <array></array></li> <li>#include <thread></thread></li> </li></ul>	Preprocessor	1	
<ul> <li>#include <ctime></ctime></li> <li>#include <vector></vector></li> <li>#include <string></string></li> <li>#include <li>limits&gt;</li> <li>#include <tuple></tuple></li> <li>#include <fstream></fstream></li> <li>#include <regex></regex></li> <li>#include <array></array></li> <li>#include <thread></thread></li> </li></ul>			
<ul> <li>#include <vector></vector></li> <li>#include <string></string></li> <li>#include <limits></limits></li> <li>#include <tuple></tuple></li> <li>#include <fstream></fstream></li> <li>#include <regex></regex></li> <li>#include <array></array></li> <li>#include <thread></thread></li> </ul>			• #include <cstdlib></cstdlib>
<ul> <li>#include <string></string></li> <li>#include <limits></limits></li> <li>#include <tuple></tuple></li> <li>#include <fstream></fstream></li> <li>#include <regex></regex></li> <li>#include <array></array></li> <li>#include <thread></thread></li> </ul>			• #include <ctime></ctime>
<ul> <li>#include <li>tuple&gt;</li> <li>#include <fstream></fstream></li> <li>#include <regex></regex></li> <li>#include <array></array></li> <li>#include <thread></thread></li> </li></ul>			• #include <vector></vector>
<ul> <li>#include <li>tuple&gt;</li> <li>#include <fstream></fstream></li> <li>#include <regex></regex></li> <li>#include <array></array></li> <li>#include <thread></thread></li> </li></ul>			#include <string></string>
<ul> <li>#include <tuple></tuple></li> <li>#include <fstream></fstream></li> <li>#include <regex></regex></li> <li>#include <array></array></li> <li>#include <thread></thread></li> </ul>			
<ul> <li>#include <fstream></fstream></li> <li>#include <regex></regex></li> <li>#include <array></array></li> <li>#include <thread></thread></li> </ul>			
<ul><li>#include <array></array></li><li>#include <thread></thread></li></ul>			•
<ul><li>#include <array></array></li><li>#include <thread></thread></li></ul>			• #include <regex></regex>
• #include <thread></thread>			
. " 1 1 2 1			
• #include <mutex></mutex>			• #include <mutex></mutex>
Pointers 1 Example of use of pointer:	Pointers	1	
• main function, line 52, playerPtr			

References	1	Example of use of references:
		Two references used in InitializePlayerData.cpp
		function
		<ul> <li>playerName and initialBalance are both passed</li> </ul>
		by reference
Function Overloading	0.5	output_DealerName function is overloaded
		One version can take strings as input
		Other version can take integers as input
Lambda Function	1	Example use of lambda function:
		Main function, line 62, lambda function name NumBetsTotal
Template Function	1	Example use of template function:
		<ul> <li>Main function, line 13, template function name</li> </ul>
		AmountWon_AndExpression
Containers	2	Used three different types of containers:
		Tuple Container in
		getBettingOptionParameters.cpp, line 4
		<ul> <li>Vector Container in main.cpp, line 97</li> </ul>
		Array Container in output_DealerName.cpp, line     5
Struct/Class	2	Three classes are created:
		Bet_Check – in Bet_Check.h
		• Bet – in Bet.h
		Player – in Player.h
Objects	1	One object per class was created
Operator Overloading	2	(+) operator, (-) operator, and (/) operator were all overloaded in the Player class found in the Player.h file
Inheritance	1	One inheritance found in Bet Check class inheriting from
mmeritanee	1	Bet class
Virtual Functions	2.	Four virtual functions can be found in Bet.h header
	_	file/class
GUI	1	Use of a text based GUI, as well as OpenSDL
File System Handling	0.5	File system handling can be found in main function.
		Project reads from "player data.txt" file and writes to the
		file as well to keep track of wins/losses by player
Regular Expression	1	Use of regular expression (win lose) found in
		EndOfGameExpression.cpp function, line 25 through 30
Concurrent Programming	1	Two threads are used in the program
		Main thread
		<ul> <li>imageThread</li> </ul>
TOTAL POINTS	30.5/35	Satisfies requirement of at least 30 points, obtains full
		credit for this section