

RA	Nota	Testes fechados corretos	Descontos relativos à qualidade do código	Comentários	Testes fechados corretos - Fora do Prazo	Descontos relativos à qualidade do código - Fora do Prazo	Comentários - Fora do Prazo	Peso do Lab	Total de casos de testes fechados
Turma A									
024925	0							2	5
090360	0								
135017	0								
148606	9,5	5	0,5	Colocou funções do torneio dentro da fila. Sempre que possível, a estrutura deve ficar isolada do problema.					
148733	0								
154781	9,5	5	0,5	Fez alguns acessos direto a posições da fila (inicio->jogador e inicio->proximo->jogador). O acesso a estrutura deve ser a partir das funções providas pelo header (fila.h).					
157510	0								
165083	0								
165117	0								
165245	10	5							
171928	9,5	5	0,5	Colocou funções do torneio dentro da fila, inclusive as funções de leitura. Sempre que possível, a estrutura deve ficar isolada do problema. Fez acessos a posições de memória inválidas no momento de destruir a fila. Sempre utilize o valgrind para verificar acessos inválidos de memória.					
176566	0								
177398	7,13				5	0,5	Fez alguns acessos direto a posições da fila (vr->ini->prox e i->prox->prox). O acesso a estrutura deve ser a partir das funções providas pelo header (fila.h).		
183020	9	5	1	Fez acessos a posições de memória não inicializadas na função "main" e não desalocou as filas. Sempre utilize o valgrind para verificar se toda a memória foi desalocada e se há acessos inválidos de memória. Acessou diretamente a estrutura (ganhadores->ini). Deveria usar as funções do header (.h) para acessar a estrutura.					
186289	0								
187679	10	5		Comente apenas o necessário. Muitos comentários deixa o código com baixa legibilidade.					
197272	0								
199415	0								
202071	0								
Turma B									
137856	2	1							
145924	9,5	5	0,5	Fez alguns acessos direto a posições da fila (principal->ini e principal->ini->prox). O acesso a estrutura deve ser a partir das funções providas pelo header (fila.h).					
147084	0								
157154	0								
160208	9,5	5	0,5	Não desalocou a memória utilizada. Sempre utilize o valgrind para verificar se toda a memória foi desalocada.					
165461	0								
167296	0								
168684	0								
170968	9,5	5	0,5	Fez alguns acessos direto a posições da fila (atual->prox->jogador). O acesso a estrutura deve ser a partir das funções providas pelo header (fila.h).					
171964	0								
172776	9	5	1	Fez alguns acessos direto a posições da fila (p_no jogador1 = repescagem->ini e p_no jogador2 = jogador1->prox). O acesso a estrutura deve ser a partir das funções providas pelo header (fila.h). Colocou funções do torneio dentro da fila. Sempre que possível, a estrutura deve ficar isolada do problema.					
173047	8,75	5	1,25	Fez alguns acessos direto a posições da fila (atual = filaPrin->inicio e atual->prox). O acesso a estrutura deve ser a partir das funções providas pelo header (fila.h). Colocou funções do torneio dentro da fila. Sempre que possível, a estrutura deve ficar isolada do problema. Não desalocou toda a memória utilizada. Sempre utilize o valgrind para verificar se toda a memória foi desalocada.					
174112	0								
175586	0								
177439	7,5	0			5				
178258	0								
178453	0								
178663	8,5	5	1,5	Fez alguns acessos direto a posições da fila (FILA->ini->prox). O acesso a estrutura deve ser a partir das funções providas pelo header (fila.h). Colocou funções do torneio dentro da fila. Sempre que possível, a estrutura deve ficar isolada do problema. Não desalocou a memória utilizada. Sempre utilize o valgrind para verificar se toda a memória foi desalocada. Comente apenas o necessário. Muitos comentários deixa o código com baixa legibilidade.					
182557	9,75	5	0,25	Não desalocou toda a memória utilizada. Sempre utilize o valgrind para verificar se toda a memória foi desalocada.					

182762	6,75	4	1,25	<p>Fez alguns acessos direto a posições da fila (aux = Principal->ini e aux->prox). O acesso a estrutura deve ser a partir das funções providas pelo header (fila.h).</p> <p>Fez acessos a posições de memória não inicializadas na função "confrontos_principal". Sempre utilize o valgrind para verificar se há acessos inválidos de memória.</p> <p>Colocou funções do torneio dentro da fila. Sempre que possível, a estrutura deve ficar isolada do problema.</p>				
184336	9	5	1	<p>Colocou funções do torneio dentro da fila. Sempre que possível, a estrutura deve ficar isolada do problema.</p> <p>Fez acessos a posições de memória não inicializadas na função "partida_repescagem" e não desalocou completamente a memória usada. Sempre utilize o valgrind para verificar se toda a memória foi desalocada e se há acessos inválidos de memória.</p> <p>Deixou trechos de código comentados.</p>				
185284	9,69	5	1,25	<p>Colocou funções do torneio dentro da fila. Sempre que possível, a estrutura deve ficar isolada do problema.</p> <p>Fez alguns acessos direto a posições da fila (TORNEIO->principal->inicio e TORNEIO->principal->inicio->prox). O acesso a estrutura deve ser a partir das funções providas pelo header (fila.h).</p>	5			
185448	0							
186291	7,5				5			
188404	6,94				5	0,75	<p>Não desalocou a memória utilizada.</p> <p>Colocou funções do torneio dentro da fila. Sempre que possível, a estrutura deve ficar isolada do problema.</p>	
193349	0							
193858	8,5	5	1,5	<p>Fez alguns acessos direto a posições da fila (aux->participante e aux->proximo->participante). O acesso a estrutura deve ser a partir das funções providas pelo header (fila.h).</p> <p>Faz remoções e adições diretamente da fila, sem usar as funções do header.</p> <p>Fez acessos a posições de memória não alocadas na função "disputa_repescagem" e não desalocou completamente a memória usada. Sempre utilize o valgrind para verificar se toda a memória foi desalocada e se há acessos inválidos de memória.</p>				
197197	0							
203308	0							
203609	8,5	5	1,5	<p>Colocou funções do torneio dentro da fila. Sempre que possível, a estrutura deve ficar isolada do problema.</p> <p>Fez alguns acessos direto a posições da fila (res [0]->jogador e res[1]->jogador). O acesso a estrutura deve ser a partir das funções providas pelo header (fila.h).</p> <p>Fez acessos a posições de memória não alocadas na função "partidas" e não desalocou completamente a memória usada. Sempre utilize o valgrind para verificar se toda a memória foi desalocada e se há acessos inválidos de memória.</p>				
205320	7,31				5	0,25	<p>Colocou funções do torneio dentro da fila. Sempre que possível, a estrutura deve ficar isolada do problema.</p>	
225162	10	5						
Turma C								
135712	9,75	5	0,25	<p>Fez alguns acessos a posições de memória não inicializadas nas funções "fila_enfileirar", "resolver" e "impar". Sempre utilize o valgrind para verificar se há acessos inválidos de memória.</p>				
138276	0							
146571	10	5						
149129	0							
163830	9	5	1	<p>Fez alguns acessos direto a posições da fila (aux->part e aux->prox->part). O acesso a estrutura deve ser a partir das funções providas pelo header (fila.h).</p> <p>Faz remoções diretamente da fila, sem usar as funções do header.</p>				
167130	10	5						
167631	10	5						
168009	0							
168633	10	5		<p>Colocou funções do torneio dentro da fila. Sempre que possível, a estrutura deve ficar isolada do problema.</p>				
171955	0							
172043	9,5	5	0,5	<p>Fez acessos a posições de memória não alocadas na função "dequeue" e não desalocou completamente a memória usada. Sempre utilize o valgrind para verificar se toda a memória foi desalocada e se há acessos inválidos de memória.</p>				
172503	0							
172580	0							
174148	0	0						
175011	0							
176216	8,75	5	1,25	<p>Colocou funções do torneio dentro da fila. Sempre que possível, a estrutura deve ficar isolada do problema.</p> <p>Fez alguns acessos direto a posições da fila (temp->prox e temp->prox->prox). O acesso a estrutura deve ser a partir das funções providas pelo header (fila.h).</p> <p>Não desalocou a memória utilizada. Sempre utilize o valgrind para verificar se toda a memória foi desalocada.</p>				
176241	0							
176746	0							
178369	0							
182255	0							
182546	0							

183360	10	5						
183930	7,5					5		
187101	10	5						
188092	6	4						
196672	0							
198487	0							
199548	9	5						
205565	8,75	5						
206156	0							
206510	7,5	0				5		
206601	0	0						