

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	6,5	4	1,5	Leitura de dados de entrada misturado com código da função realoca (menos 0,5 pontos). A memória não está sendo apropriadamente desalocada; por exemplo, a função destroiLista só desaloca a memória referente ao primeiro elemento da lista (menos 0,5 pontos). Programa acessa posições não alocadas de memória (menos 0,5 pontos).					
148733	0								
154781	10	5							
157510	0								
165083	0								
169117	0								
165245	10	5		Para melhor garantir a correteza do programa, é aconselhável que a função destruir_lista seja capaz de alterar o ponteiro da lista para NULL , para que ele não continue apontando para uma região de memória já desalocada (sem desconto na pontuação).					
171928	0	0							
176566	0								
177398	7,5				5				
183020	0								
186289	0								
187679	10	5							
197272	0								
199415	0								
202071	0								
Turma B									
137856	0	0							
145924	0								
147084	0								
157154	0								
160208	0								
165461	0								
167296	3	2	1	Memória não está sendo apropriadamente desalocada; para todos os testes, o valgrind aponta regiões de memória não desalocadas (menos 0,5 pontos). A manipulação da lista ligada não está sendo realizada no módulo lista ; ao invés disso, o código responsável pela manipulação da lista está misturado com o código responsável por gerenciar informações sobre a memória (dentro do módulo memoria) (menos 0,5 pontos).					
168684	0								
170968	10	5							
171964	0								
172776	4,8	3	1,2	Os arquivos cabeçalhos não utilizam a estrutura básica (https://bit.ly/2v9j6wC) que um arquivo cabeçalho deve ter (menos 0,2 pontos). A memória não está sendo apropriadamente desalocada (menos 0,5 pontos). Eventualmente, o programa acessa posições não alocadas de memória (menos 0,5 pontos). O arquivo cabeçalho memoria.h não utiliza a estrutura básica (https://bit.ly/2v9j6wC) que um arquivo cabeçalho deve ter (menos 0,1 pontos). A memória não está sendo apropriadamente desalocada; por exemplo, no teste arg03.in , houveram 14 alocações e apenas 4 liberações de memória alocada (menos 0,5 pontos). Operações responsáveis pelo gerenciamento da memória estão sendo realizadas no módulo lista ao invés de no módulo memoria ; por exemplo, a função adicionar_na_lista está realizado a operação de desalocar; no geral, os dois módulos não estão bem separados (menos 0,5 pontos).					
173047	8,9	5	1,1						
174112	6,75					5	1	A memória do programa não está sendo devidamente desalocada; por exemplo, o módulo lista.h não contém nenhuma função responsável por eliminar um elemento da lista (menos 0,5 pontos). A remoção de um elemento da lista (sem a liberação da memória) está sendo realizado dentro do módulo memoria.h ao invés de ser realizado dentro do módulo lista.h , ou seja, há uma confusão entre os conceitos de memória e lista (menos 0,5 pontos).	
175586	0								
177439	0								
178258	0								
178453	0								
178663	9	5	1	O caractere \n contido em cada linha de entrada para o programa não está sendo apropriadamente lido; utilizou-se contorno inapropriado ao invés de solucionar o problema (menos 0,5 pontos). Leitura de dados de entrada misturado com código da função realocar (menos 0,5 pontos). Os arquivos cabeçalhos não utilizam a estrutura básica (https://bit.ly/2v9j6wC) que um arquivo cabeçalho deve ter (menos 0,2 pontos). A memória não está sendo apropriadamente desalocada; por exemplo, não foi implementada nenhuma função responsável por destruir a lista ligada (menos 0,5 pontos).					
182557	9,3	5	0,7	Os arquivos cabeçalhos não utilizam a estrutura básica (https://bit.ly/2v9j6wC) que um arquivo cabeçalho deve ter (menos 0,2 pontos). A memória não está sendo apropriadamente desalocada (menos 0,5 pontos). Funções dos módulos lista e memoria estão misturadas entre os arquivos lista.h e memoria.h , ou seja, as operações sobre a lista ligada não estão separadas das operações de gerenciamento de memória (menos 0,5 pontos).					
182762	0,8	1	1,2	Leitura de string em uma região de memória alocada como um único caractere (o scanf atribui um caractere \0 no final de toda string lida) (menos 0,2 pontos). Operação de remoção de elemento da lista não foi apropriadamente implementada no módulo lista ; ao invés, operação está sendo realizada no módulo memoria (menos 0,5 pontos).					
184336	9,3	5	0,7						

185284	9,75	5	1	Os arquivos cabeçalhos não utilizam a estrutura básica (https://bit.ly/2V9j9wC) que um arquivo cabeçalho deve ter (menos 0,2 pontos). Código demasiadamente confuso e carente de comentários (menos 0,3 pontos). Há diversas operações sobre a lista ligada que são realizadas na função <code>main</code> ao invés de serem apropriadamente implementadas no módulo <code>lista</code> (menos 0,5 pontos).	5	
185448	0					
186291	0					
188404	3	2	1	Não foram implementadas as funções que devem manipular a lista ligada, como, por exemplo, funções responsáveis por inserir e remover elementos; ao invés, estas operações estão sendo realizadas misturadas com as funções do módulo <code>memoria</code> (menos 0,5 pontos). A memória utilizada pelo programa não está sendo apropriadamente desalocada; por exemplo, não foi implementada uma função responsável por desalocar a memória utilizada pela lista ligada (menos 0,5 pontos).		
193349	0					
193858	0					
197197	0					
203308	0					
203609	0	5	10	O programa aloca demasiada memória para rodar (usa-se um <code>malloc</code> para cada bloco de memória que o programa deve simular); para isso, utiliza-se um campo (<code>status</code>) no <code>struct</code> utilizado para marcar se a região de memória está sendo utilizada; em resumo, programa foi implementado de modo que informações sobre blocos de memória alocados também são manipulados, ou seja, não satisfaz requisito básico do enunciado (menos 10,0 pontos).		
205320	0	0				
225162	0					
Turma C						
135712	0					
138276	0					
146571	9,5	5	0,5	A região de memória alocada para o <code>struct Lista</code> não é desalocada até o término da execução do programa (menos 0,2 pontos). Apesar das operações dos módulos <code>lista</code> e <code>memoria</code> estarem bem separadas em seus respectivos módulos, não foi implementada uma função para remover um elemento da lista; esta operação está sendo realizada no módulo <code>memoria</code> (menos 0,3 pontos).		
149129	0					
163830	8,8	5	1,2	Leitura de dados de entrada misturado com código da função <code>realloc</code> (menos 0,5 pontos). Nomes contraintuitivos dado às variáveis, como, por exemplo, <code>aux</code> e <code>aux2</code> (menos 0,2 pontos). Não foram implementadas as funções que devem manipular a lista ligada, como, por exemplo, funções responsáveis por inserir e remover elementos; ao invés, estas operações estão sendo realizadas misturadas com as funções do módulo <code>memoria</code> (menos 0,5 pontos).		
167130	10	5				
167631	9,6	5	0,4	Leitura de string em uma região de memória alocada como um único caractere (o <code>scanf</code> atribui um caractere <code>\0</code> no final de toda string lida) (menos 0,2 pontos). Nomes contraintuitivos dado às variáveis, como, por exemplo, <code>aux</code> e <code>auxe</code> (menos 0,2 pontos).		
168009	0					
168633	9	4	0,5	A memória utilizada pelo programa não está sendo apropriadamente desalocada; por exemplo, a função <code>excluir_no</code> não faz chamada alguma à função <code>free</code> (menos 0,5 pontos).	5	0,5
171955	0					
172043	0					
172503	0					
172580	0					
174148	0					
175011	0					
176216	3,5	2	0,5	Declaração da mesma estrutura tanto no arquivo <code>lista.c</code> quanto no arquivo <code>lista.h</code> ; o arquivo <code>lista.c</code> não incluiu (não implementou) as funções declaradas em <code>lista.h</code> (menos 0,5 pontos).		
176241	0					
176746	0					
178369	0					
182255	0					
182546	0					
183360	0	4	10	O programa aloca demasiada memória para rodar (usa-se um <code>malloc</code> para cada bloco de memória que o programa deve simular); para isso, utiliza-se um campo (<code>alocado</code>) no <code>struct</code> utilizado para marcar se a região de memória está sendo utilizada; em resumo, programa foi implementado de modo que informações sobre blocos de memória alocados também são manipulados, ou seja, não satisfaz requisito básico do enunciado (menos 10,0 pontos).		
183930	7,13	0			5	0,5
187101	9,9	5	0,1	Má formatação no <code>struct no</code> declarado no arquivo <code>lista.h</code> (menos 0,1 pontos). O programa aloca demasiada memória para rodar (usa-se um <code>malloc</code> para cada bloco de memória que o programa deve simular); para isso, utiliza-se um campo (<code>estado</code>) no <code>struct</code> utilizado para marcar se a região de memória está sendo utilizada; em resumo, programa foi implementado de modo que informações sobre blocos de memória alocados também são manipulados, ou seja, não satisfaz requisito básico do enunciado (menos 10,0 pontos).		
188092	0	2	10			
196672	0					
198487	0					
199548	9,3	5	0,7	Má formatação (alinhamento incorreto) em diversas partes pontuais do código (menos 0,2 pontos). Eventualmente se faz uso (por exemplo, na função <code>imprimir_lista</code>) de valor não inicializado do <code>struct no</code> : <code>prox</code> e <code>ant</code> (menos 0,5 pontos).		

				O programa aloca demasiada memória para rodar (usa-se um malloc para cada bloco de memória que o programa deve simular); para isso, utiliza-se um campo (seg) no struct utilizado para marcar se a região de memória está sendo utilizada; em resumo, programa foi implementado de modo que informações sobre blocos de memória alocados também são manipulados, ou seja, não satisfaz requisito básico do enunciado (menos 10,0 pontos).					
205565	0	2	10						
206156	0								
206510	0								
206601	0	0							