



6 Busca Binária



(+++)

Considere que seja dado um vetor v de números naturais considerados como *chaves primárias*. Sobre este vetor deseja-se realizar a operação de pesquisa (ou busca) por uma determinada destas chaves. Um método eficiente para realização desta pesquisa é denominado de “Busca Binária”. Entretanto ele exige que o vetor esteja ordenado.

Seguindo as especificações a seguir, você deve implementar um algoritmo *recursivo* para realizar a busca binária em v .

Entrada

Na primeira linha será fornecido o número natural t , $1 \leq t \leq 10$, que representa o número de casos de testes.

Cada caso de teste possui duas linhas de entrada: (1) a primeira contém o valor da chave, k , a ser pesquisado no vetor v ; (2) a segunda possui os elementos de v , da primeira posição em direção à última. Para indicar a finalização da entrada dos dados do vetor, o valor -1 (menos um) é utilizado como *flag*.

Tanto os elementos de v quanto a chave de pesquisa k variam no intervalo de 1 a $(2^{32} - 1)$, inclusive extremos.

Saída

Seu programa deve imprimir, para cada caso de teste, uma linha contendo a posição em que a chave k foi encontrada no vetor v ou, no caso de sua ausência, a palavra “ausente” (com todas as letras minúsculas).

Exemplos

Entrada	Saída
3	6
8	1
1 7 9 4 2 8 3 5 6 10 -1	ausente
1	
1 7 9 4 2 8 3 5 6 10 -1	
14	
1 7 9 4 2 8 3 5 6 10 -1	

Entrada	Saída
3	6
81	10
18 71 93 44 25 81 13 15 61 100 -1	ausente
108	
101 711 19 224 235 889 389 511 619	
108 -1	
141	
11 17 91 24 652 18 31 587 601 110 -1	

Observação: Apenas a implementação que utilize *recursividade* será considerada válida como resposta, ou seja, uma implementação iterativa, mesmo que com todas as saídas corretas, não será aceita.