

## Grades

Um de seus professores cometeu um terrível erro ao salvar nas notas da turma em um CSV, onde cada coluna representa um aluno e cada linha uma avaliação. O professor esqueceu das vírgulas, tornando o CSV um VVV e totalmente ilegível. Como ele já tinha demorado dois meses para organizar as notas e você não quer esperar mais, faça um programa, obedecendo o cabeçalho *grades.h*, receba como entrada o arquivo VVV, o número de alunos e retorne as medias de cada aluno. Tenha em mente que cada nota é um inteiro, já que esse professor só aplica provas objetivas contendo dez questões, e que as notas estarão entre 1 e 10, pois ele nunca dá nota zero pra nenhum aluno.

## MergeMorse

Você está no meio da terceira mundial, mas seu professor de Cálculo passou alguns exercícios que precisam ser resolvidos e, como estamos sobre uma ameaça de bombardeio, você está escondido no abrigo anti-bombas na sua casa e você resolve fazer os exercícios propostos, mas como eles se mostraram muito difíceis, você decide consultar seus colegas sobre a resolução da lista. Com a ajuda de um velho rádio e utilizando código morse, você decide falar com seus colegas de classe para que alguém possa lhe ajudar, utilizando a seguinte tabela:

Simbolo	Valor
.	1
-	5
:	2
=	10

Todos os exercícios propostos que você não conseguiu resolver contém apenas operações de adição e multiplicação (por exemplo “=-..+...\*:. ” representando “18+3\*3” que é igual a 27). Para não ter que ficar calculando as respostas na mão você decide fazer um programa recursivo, obedecendo o cabeçalho *merge\_morse.h* que, dado uma string contendo um código morse que representa uma expressão válida, retorna o seu resultado.

## Tornado!

Após um violento tornado, a cerca de um fazendeiro ficou parcialmente destruída, e ele pediu sua ajuda para concertá-la. A cerca era construída de maneira uniforme de postes de concreto a cada dois metros. Como pode demorar um pouco para se iniciarem os reparos, o fazendeiro quer colocar postes de madeira no lugar dos de concreto para poder fechar a cerca e, visando gastar o mínimo possível, ele decidiu colocar os postes de madeira tal que a distância entre postes consecutivos seja no máximo de quatro metros. Faça um programa, obedecendo o cabeçalho *tornado.h*, que receba o número de postes, um array binário representando suas posições, onde o valor 0 significa poste a ser substituído e 1 poste a ser mantido e retorne o número de postes mínimos necessários.

## Hanói

Um clássico! Ele não poderia faltar aqui! As regras são as conhecidas: dados  $n$  discos de tamanhos distintos e três postes você deve mover todos os discos de um poste para qualquer outro onde durante o percurso nenhum disco maior fique sobre um menor. Faça um programa, obedecendo o cabeçalho *hanoi.h*, que receba o número de discos e retorne o número de trocas realizadas para solucionar o problema.

## ***Bakugan***

Você e um amigo seu estão jogando um jogo baseado em Bakugan, e o jogo consiste no seguinte: existem 10 monstros diferentes, cada um ranqueado de acordo com o seu poder. O jogo é jogado em R rounds e, em cada round, cada jogador joga um monstro aleatório e acumula pontos de acordo com o ranking do monstro escolhido. Porém existe uma regra especial: o primeiro, e apenas o primeiro jogador que jogar o mesmo monstro durante três rounds consecutivos ganha 30 pontos extras. Porém, caso ambos realizem esse feito na mesma rodada ninguém ganha os pontos. Faça um programa, obedecendo o cabeçalho *bakugan.h* que, dado o número de rounds e os monstros escolhidos em cada round anuncie o vencedor na forma de um inteiro:

- 0: empate;
- 1: você ganhou;
- -1: seu amigo ganhou.

## ***Put The Floor***

Cansado do mundo estritamente computacional você decide se aventurar no ramo da construção civil e seu primeiro trabalho é arrumar o piso de um salão retangular. Para isso você possui diversas tábuas de diversos comprimentos e em quantidades variadas. Porém antes de começar, você percebe que esqueceu seu serrote em casa, então terá que usar as tábuas sem serrá-las. Para sua sorte, você comprou todas as tábuas com a mesma largura, tendo apenas uma variação no comprimento das mesmas. Além disso, o próprio dono do salão impôs duas restrições para que seus serviços fossem contratados:

- Todas as tábuas devem estar no mesmo sentido;
- Cada linha do piso deve conter, no máximo, duas tábuas.

Como uma vez programador sempre programador, você decide fazer um programa, obedecendo o cabeçalho *put\_the\_floor.h*, que recebe de entrada as dimensões do salão, a largura e o comprimento das tábuas, o número de tábuas de cada comprimento e retorne quantas tábuas são necessárias ou retorne -1 caso seja impossível.

## ***Power Plant***

Um engenheiro eletricista está com um problema: ele montou uma rede de distribuição entre n cidades, mas essa rede está com um custo de implementação muito alto. Analisando o projeto, ele constatou que a maior parte do orçamento é destinada a compra dos cabos de transmissão. Faça um programa, obedecendo o cabeçalho *power\_plant.h*, que dado uma malha de transmissão, representada na forma de um grafo, e os custos de implementação de cada ligação (peso nas arestas do grafo), retorne uma malha onde a partir de todos os nós exista um caminho que os leva a todos os outros e que essa malha tenha custo de implementação mínimo.

## ***Give me some change***

Você foi a um supermercado fazer as compras do mês e, após tudo selecionado, se dirigiu ao caixa para efetuar o pagamento, mas o caixa que escolheu está em treinamento. A garota do caixa, por querer fazer uma boa impressão nos clientes, insiste em dar o troco utilizando o menor número de moedas possível, mas a coitadinha vive se perdendo nas contas e acaba por começar tudo de novo. Faça um programa obedecendo o cabeçalho *give\_me\_some\_change.h*, que recebe como entrada os tipos de moedas disponíveis (suponha que são infinitas e a moeda de 1 centavo sempre está entre elas), e retorne a quantidade de moedas de cada tipo que a garota deve retornar. Extra: talvez seu programa demore um pouco, refaça o programa agora com um cabeçalho a sua escolha, isso pode ajudar.

## ***How many piles?***

Vida de universitário não é mole e por isso você resolve ganhar um dinheiro cuidando do filho de um vizinho seu. Chegando lá, o garoto propõe um jogo: dadas  $n$  moedas, cada jogador deve organizar essas moedas em até  $n$  pilhas de formas diferentes de todas as outras já propostas, desconsiderando permutações das pilhas. Faça um programa, obedecendo o cabeçalho *how\_many\_piles.h*, que recebe o número de moedas e retorna o número de permutações possíveis.

## ***The Lazy Runner***

Você decide entrar em uma prova física, que consiste no seguinte: cada participante recebe um mapa, no formato que escolher, e uma lista de pontos no mapa nos quais são obrigatórios a passagem, na ordem estipulada. Como você é um superprogramador, e superpreguiçoso, você resolve fazer um programa, obedecendo o cabeçalho *the\_lazy\_runner.h* que, dado um grafo (seu mapa), uma lista de vértices desse grafo, os pontos obrigatórios, e o ponto de início, descobre o menor percurso para terminar essa prova.

## ***Critical Route***

Uma tragédia aconteceu recentemente em sua cidade. Um paciente em condição crítica, que necessitava tratamento urgente, morreu enquanto era transportado para um grande hospital da capital do estado. O que ocorreu foi que a ambulância ficou presa no trânsito, devido a uma rocha que deslizou na estrada. A população reclamou com o governador, que agora deseja evitar acontecimentos similares no futuro. Infelizmente, deslizamentos de rochas são muito comuns nesse estado, com muitas montanhas e serras. Assim, para minimizar o número de tragédias devidas a deslizamentos de rochas e outros imprevistos, o governador decidiu criar rotas alternativas entre cada cidade do estado e a capital. Para isso, é necessário inicialmente identificar quais segmentos de estradas são atualmente críticos, isto é, se bloqueados causam que não haja caminho possível entre alguma cidade e a capital. Um segmento de estrada é um trecho de estrada que liga duas cidades distintas. Sua tarefa é escrever um programa, obedecendo o cabeçalho *critical\_route.h*, para identificar esses segmentos críticos de estradas.

## ***Copa do Mundo***

Uma Copa do Mundo de futebol de botões está sendo realizada com times de todo o mundo. A classificação é baseada no número de pontos ganhos pelos times, e a distribuição de pontos é feita da forma usual. Ou seja, quando um time ganha um jogo, ele recebe 3 pontos; se o jogo termina empatado, ambos os times recebem 1 ponto; e o perdedor não recebe nenhum ponto. Faça um programa obedecendo o cabeçalho *world\_cup.h* que receba o número de times participantes, o número de partidas jogadas, as pontuações dos mesmos e retorne quantos empates ocorreram na Copa.

## ***Sort***

O problema de ordenação é um problema clássico, assim existem dezenas de algoritmos para resolvê-lo, cada um seguindo uma filosofia, para não ficarmos apenas nos básicos faça três algoritmos de sort, de preferência três que você não conheça, que ordenam arrays genéricos, seguindo cabeçalho *sort.h*, e compare sua performance.