

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 ele esqueceu das vírgulas tornado o CSV em 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 e o número de alunos e retorne as medias de cada aluno, tenha em mente que cada nota é um inteiro, esse professor só passa provas objetivas de dez questões ♥, e que estejam entre 1 e 10, ele nunca dá nota zero pra nenhum aluno ♥♥.

MergeMorse

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

Simbolo	Valor
.	1
-	5
:	2
=	10

Todos os exercícios propostos que você não conseguiu resolver contem apenas operações de adição e multiplicação, esse não é o seu melhor dia, assim um exemplo de expressão seria “=-...+...*:.” que representa “18+3*3” que é igual a 27, lembre-se da precedência dos operadores. 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 valida e retorna o seu resultado. E sim, tem um computador no abrigo e você está se comunicando com seus colegas por uma rádio usando código morse em pleno século XXI resolvendo expressões aritméticas básicas que foram deixadas por um professor de cálculo, mas ninguém disse que a vida tinha que fazer sentido, não é mesmo?

Tornado!

Após um violento tornado, a cerca de um fazendeiro ficou parcialmente destruída, sim isso mesmo, milhares de pessoas desabrigadas, mas estamos interessados apenas na cerca do fazendeiro, 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, talvez pelo fato da cidade estar destruída, mas não estamos interessados nisso, o fazendeiro quer colocar postes de madeira no lugar nos de concreto para poder fechar a cerca, mas como ele quer gastar o mínimo possível, além de insensível é avaro, 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 recebe o número de postes, um array binário representando-os, onde o valor 0 significa poste a ser substituído e 1 poste a ser mantido e retorne o número de postes mínimos necessários.

Hanoi

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 recebe o número de discos e retorna o número de trocas realizadas para solucionar o problema.

Bakugan

Você e um amigo seu estão jogando um jogo baseado em Bakugan, e ambos possuem péssimo gosto, mas isso não é da minha conta, 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, em cada round cada jogador joga um monstro aleatório, cada jogador acumula pontos de acordo com o ranking do monstro jogado, o primeiro, e apenas o primeiro, jogador que jogar o mesmo monstro durante três rounds consecutivos ganha 30 pontos extras, caso ambos realizem esse feito na mesma rodada ninguém ganha os pontos. Como você é ruim de matemática, veja o problema acima caso tenha dúvidas disso, você decide fazer um programa, obedecendo o cabeçalho *bakugan.h* que dado o número de rounds e os monstros escolhidos em cada round anuncia 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, seu primeiro trabalho é arrumar o piso de um salão retangular, para isso você possui diversas tábuas de diversos comprimentos e em quantidades variadas, várias restrições foram impostas pelo dono do salão:

- Todas as tábuas estejam no mesmo sentido;
- Cada linha do piso contenha no máximo duas tábuas;
- As tábuas não podem ser serradas.

Como uma vez programador sempre programador, você decide fazer um programa, obedecendo o cabeçalho *put_the_floor.h*, recebe de entrada as dimensões do salão, a largura das tábuas e seu comprimento e 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, ele analisou o projeto e 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, 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_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 retorna 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, por isso você resolve ganhar um dinheiro cuidando do filho de um vizinho seu, o garoto propõe um jogo: dado n moedas, cada jogador deve organizar essas moedas em até n pilhas de formas diferente de todas as outras já propostas, desconsiderando permutações das pilhas. Você procura não aceitar esse jogo enfadonho, mas o garoto começa a chorar se você disser não pra ele, então você resolve fazer um programa, obedecendo o cabeçalho *how_many_piles.h*, que recebe o número de moedas e retorna o número de permutações possíveis para que você tenha uma ideia de quanto tempo da sua vida será jogado fora com esse jogo.

The Lazy Runner

Você decide entrar em uma prova física, só deus sabe o porque, a prova é a 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.