

1 Sobre a entrega do trabalho

São requisitos para atribuição de notas a este trabalho:

- Uso de um arquivo **makefile** para facilitar a compilação. Os professores rodarão “make” e deverão obter o arquivo executável funcional com a sua solução. Este executável deverá estar no subdiretório tp4e deve ter o nome (**tp4**);
- Opções de compilação: deve incluir **-Wall** e **-std=c90**. Haverá desconto na nota se compilador mostrar algum “warning”;
- Seu programa deve desalocar (**free**) toda a memória dinâmica alocada, isto será conferido pelo uso do programa **valgrind**, cada erro também vai gerar desconto na nota;
- Arquivo de entrega:
 - deve estar no formato tar comprimido e deverá obrigatoriamente ter o nome (**tp4.tar.gz**);
 - O **tar** deve ser criado considerando-se que existe um diretório com o nome do trabalho. Por exemplo, este trabalho é o tp4;
 - Então seu **tar** deve ser criado no diretório pai do subdiretório tp4, o qual deve conter todos os arquivos que serão entregues (tar zcvf tp4.tar.gz tp4), de maneira que os professores, ao abrirem o **tar** com o comando “tar zxvf tp4.tar.gz” obterão um diretório tp4 com as suas respostas;
 - Os professores testarão seus programas em uma máquina do departamento de informática (por exemplo, cpu1), por isso, antes de entregar seu trabalho faça um teste em máquinas do dinf para garantir que tudo funcione bem.

2 Objetivos

São objetivos deste trabalho a prática dos seguintes conceitos:

- Tipos abstratos de dados
- Alocação dinâmica de structs e de vetores;
- Manipulação de ponteiros;
- Uso da ferramenta `valgrind`.

3 O problema

Um grupo de super-heróis recebeu um chamado e precisa montar uma equipe para ir resolver o problema. Cada um dos membros do grupo tem uma série de poderes, como super força, super velocidade, super visão etc. O chamado indicava uma lista de habilidades que uma equipe de super-heróis deveria ter para ser enviada e resolver o problema com sucesso.

Caso uma habilidade não seja contemplada neste conjunto de heróis a missão falhará. O grupo de super-heróis preparou uma lista de possíveis equipes. Nossa tarefa, como parte da equipe de TI do grupo, é encontrar qual destas possíveis equipes satisfaz os critérios e tem o menor tamanho.

Você deve ler, inicialmente, uma lista de habilidades para cada super-herói. Os super-heróis são numerados, a partir de 1. As habilidades são números inteiros. Uma lista de habilidades termina com um 0 (zero). A entrada, desta parte, consistirá de uma sequência de listas de habilidades, terminando com uma lista vazia (que não é de nenhum super-herói).

Após esta parte, você deve ler os requisitos da missão, que também são descritos por uma lista de habilidades, nos mesmos moldes da parte anterior.

Em seguida você deve ler uma lista de equipes, onde cada equipe também é uma sequência de números (de cada super-herói) terminada com 0 (zero). A lista de equipes termina com uma equipe vazia.

Como saída você deve escrever a lista dos super-heróis da equipe que melhor satisfaz os requisitos. Caso nenhuma equipe satisfaça, você deve escrever “NENHUMA”.

Nunca imprima espaços em branco após o término de uma linha.

Exemplo de entrada:

```
1 2 3 4 0           // super-herói 1
2 4 8 20 17 0       // super-herói 2
99 5 12 6 0         // super-herói 3
1000 0              // super herói 4
4 9 3 99 0          // super-herói 5
0
```

```

4 99 1000 0          // habilidades requeridas
1 2 3 0              // equipe 1
1 2 3 4 5 0          // equipe 2
2 4 5 0              // equipe 3
5 0                  // equipe 4
4 5 0                // equipe 5
1 2 5 0              // equipe 6
0

```

Saída correspondente:

```
4 5
```

Lembrando que após o último número impresso não há espaços em branco antes do `\n`.

4 Seu programa

Você vai receber junto com este enunciado um arquivo `tp4.tar.gz` o qual depois de aberto conterá quatro arquivos, sendo um deles o arquivo `lib_conjunto.h` que deve ser utilizado como base para a implementação do seu `lib_conjunto.c`.

Você também receberá mais dois outros arquivos:

- `testa_conjunto.c`, para você testar sua lib;
- `tp4.c`, um rascunho de implementação da solução do problema dos vingadores.

O arquivo `lib_conjunto.h` não poderá ser modificado por você, na correção os professores usarão o mesmo arquivo que foi disponibilizado. Seu arquivo `lib_conjunto.c` deve ser implementado considerando os protótipos na construção dos outros dois arquivos. Note que as funções devem funcionar exatamente como está escrito nos comentários delas.

Caso você precise, ou queira, usar outras funções, estas devem estar localizados em um dos outros dois arquivos, a escolha do local correto faz parte da avaliação.

Você também vai receber uma sugestão de arquivo `makefile` e um arquivo que testa sua implementação dos conjuntos. Este arquivo pode ser usado para você validar a biblioteca antes de implementar o `tp4.c` propriamente dito.

Considere que o um super-herói nunca tem mais do que 10 habilidades.

Seu arquivo `tp4.c` deve usar a sua `lib_conjunto.h` para resolver o problema descrito acima.

5 Referências bibliográficas

Para fazer este trabalho você deve estudar o conceito de tipos abstratos de dados, em particular o TAD conjunto.

Uma referência para o TAD conjunto é o livro de alg1, no capítulo 12, no qual você encontrará o referencial teórico e duas sugestões de implementação deste TAD escritas em Pascal. Faça as adaptações necessárias. Aproveite para estudar as complexidades dos algoritmos implementados nas duas versões e escolha a que mais lhe parece eficiente no contexto geral.

O material pode ser encontrado aqui:

https://www.inf.ufpr.br/marcos/livro_alg1/livro_alg1.pdf.

Outra referência excelente para tipos abstratos de dados é o livro do Cormem e colegas, indicado na ficha 2 da disciplina.

Bom trabalho!