



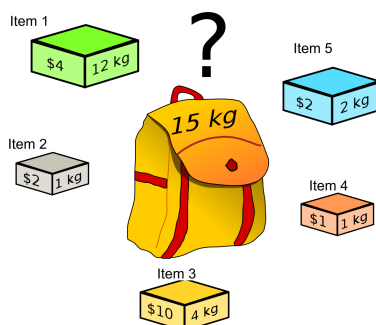
### Trabalho Prático 2 – Valor 15 pontos!

Neste trabalho prático você irá demonstrar suas habilidades de resolver e interpretar um problema computacionalmente complexo chamado Mochila 0-1, também chamado de Mochila Binária (em inglês, *Knapsack problem*). Em tal exercício, dentre outras, você irá mostrar suas habilidades com implementação de funções, passagens de parâmetros por referência, tipo abstrato de dados, modularização e tipos de dados heterogêneos.

De maneira resumida, suponha que, dado um conjunto de objetos (itens), cada um com um certo peso (*Kg*) e um certo valor (benefício \$). Sabendo que a mochila tem capacidade limitada, quais dos objetos devo colocar na minha mochila para que o valor total seja o maior possível? (Duvidas veja a gravação da aula que falamos deste assunto)

Como percebido, o problema da mochila é um problema combinatório que possui diferentes soluções. Sabendo que  $n$  é o número de possíveis de itens a serem inseridos na mochila, e como cada item pode ser colocado ou não na mochila, o número de soluções é exponencial, sendo de  $2^n$ . No exemplo abaixo, como existem 5 itens, o número de possíveis soluções é  $2^5 = 2 \times 2 \times 2 \times 2 \times 2 = 32$ . Perceba também que existem soluções viáveis e inviáveis, por exemplo:

- Colocar na mochila os itens: 1(verde), 2(cinza) e 3(amarelo) gera um benefício de  $\$4 + \$2 + \$10 = \$16$ , mas é uma solução **inviável**, pois possui o peso de  $12Kg + 1Kg + 4Kg = 17Kg$ , o que extrapola a capacidade da mochila que é de  $15Kg$ ;
- Colocar na mochila os itens: 2(cinza), 3(amarelo), 4(laranja) e 5(azul) gera um benefício de  $\$2 + \$10 + \$1 + \$2 = \$15$ , e é uma solução **viável**, pois possui o peso de  $1Kg + 4Kg + 1Kg + 2Kg = 8Kg$ , o que **não** extrapola a capacidade da mochila que é de  $15Kg$ ;



A maneira mais usual de representar uma solução para o problema da Mochila Binária é através de vetores binários, onde cada posição do vetor pode representar um item da mochila, e se este item for inserido na solução, ele irá ser assinalado com o valor 1, e caso o item não faça parte da solução, ele será assinalado com o valor 0. Exemplo:

- Colocar na mochila os itens: 1(verde), 2(cinza) e 3(amarelo):

	Item 1	Item 2	Item 3	Item 4	Item 5
Benefício de cada item	\$4	\$2	\$10	\$1	\$2
Peso de cada item	12Kg	1Kg	4Kg	1Kg	2Kg
Vetor de soluções binário	1	1	1	0	0

Benefício = \$16  
Peso = 17Kg  
Inviável

- Colocar na mochila os itens: 2(cinza), 3(amarelo), 4(laranja) e 5(azul) :

	Item 1	Item 2	Item 3	Item 4	Item 5
Benefício de cada item	\$4	\$2	\$10	\$1	\$2
Peso de cada item	12Kg	1Kg	4Kg	1Kg	2Kg
Vetor de soluções binário	0	1	1	1	1

Benefício = \$15  
Peso = 8Kg  
Viável

O problema é bem simples, mas tem muito serviço pela frente. Sua equipe foi contratada para resolver este problema de diferentes maneiras, sendo:

- Escolha de itens aleatória com várias partidas. Gere o vetor solução dentro de uma estrutura de repetição, gere 5 mil soluções aleatórias, e salve a melhor;
- Heurística de maior benefício. Escolha os itens a serem inseridos iniciando pelos de maior benefício;
- Heurística de menor peso. Escolha os itens a serem inseridos iniciando pelos de menor peso;
- Heurística de melhor relação peso benefício. Escolha de itens iniciando pela melhor relação peso benefício.
- Força bruta (analisa todas as soluções existentes, garante a solução ótima);

### Etapas do trabalho prático:

Neste trabalho prático você deve realizar as ações **nesta ordem**:

1. Implemente o código em módulos, separados em arquivos “.h” e “.c”.
2. Cada função criada deve ser obrigatoriamente explicada por comentários no código. Deve-se explicar os parâmetros de entrada, e o retorno das funções, bem como seu funcionamento. Assim, descreva detalhadamente, com comentários, o que cada função faz! Cuidado, se você entender a função errado, e descrever errado, você irá perder pontos! Certifique-se que entendeu e comentou corretamente cada função!
3. Confira se as saídas do Força Bruta estão gerando o resultado esperado, conforme exemplo no final deste documento.
4. Efetue a etapa de experimentação, conforme explicado, e gere um relatório em forma de gráficos com os resultados obtidos. Os dados da experimentação devem ser lidos de arquivos a serem disponibilizados pelo professor.

#### Etapa 4 - Confira as saídas se estão gerando o resultado esperado.

Se as funções anteriores foram implementadas corretamente para as seguintes entradas, devem ser geradas as seguintes saídas! Repare que a abordagem de Força Bruta, inserção por menor peso, e inserção por maior benefício devem gerar saídas iguais às suas, pois são algoritmos determinísticos. Por outro lado, as abordagens aleatórias podem gerar saídas diferentes, pois como dito, são aleatórias. Os tempos de execução também podem variar, visto que estes são dependentes da capacidade de processamento do computador que você está utilizando.

**Teste 1 de verificação – 5 itens (figura inicial):**

```
Capacidade da mochila 15
Item1 Item2 Item3 Item4 Item5
vetor de peso [ 12 1 4 1 2]
vetor de beneficio [ 4 2 10 1 2]

Melhor solucao Forca Bruta (otima)
vetor solucao: [01111] peso 8 beneficio 15 Tempo de execucao: 0.000000 ms

Solucao inserindo item de menor peso
vetor solucao: [01111] peso 8 beneficio 15 Tempo de execucao: 0.000000 ms

Solucao inserindo item de maior lucro
vetor solucao: [01111] peso 8 beneficio 15 Tempo de execucao: 0.000000 ms

Melhor solucao Aleatoria em 12500 iteracoes
vetor solucao: [01111] peso 8 beneficio 15 Tempo de execucao: 3.000000 ms

Melhor solucao Aleatoria em 25000 iteracoes
vetor solucao: [01111] peso 8 beneficio 15 Tempo de execucao: 4.000000 ms

Melhor solucao Aleatoria em 50000 iteracoes
vetor solucao: [01111] peso 8 beneficio 15 Tempo de execucao: 8.000000 ms
```

**leitura do arquivo:**

```
15 Limite de peso
5 Quantidade de itens
12 1 4 1 2 Peso de cada item
4 2 10 1 2 Beneficio de cada item
```

**Arquivo disponível no link:**

[https://drive.google.com/file/d/1hxBPU7mNLn2QaoEp17ZPjtDUu\\_0N8q1s/view?usp=sharing](https://drive.google.com/file/d/1hxBPU7mNLn2QaoEp17ZPjtDUu_0N8q1s/view?usp=sharing)

**Teste 2 de verificação – 15 itens:**

Capacidade da mochila	20	Item1	Item2	Item3	Item4	Item5	Item6	Item7	Item8	Item9	Item10	Item11	Item12	Item13	Item14	Item15
vetor de peso	[	4	5	2	7	9	4	5	3	5	8	1	3	5	5	7]
vetor de beneficio	[	3	6	1	3	8	3	6	4	2	4	7	5	4	6	7]

Melhor solucao Forca Bruta (otima)  
vetor solucao: [010000100011010] peso 19 beneficio 30 Tempo de execucao: 164.000000 ms

Solucao inserindo item de menor peso  
vetor solucao: [101001010011000] peso 17 beneficio 23 Tempo de execucao: 0.000000 ms

Solucao inserindo item de maior lucro  
vetor solucao: [000010000011001] peso 20 beneficio 27 Tempo de execucao: 0.000000 ms

Melhor solucao Aleatoria em 12500 iteracoes  
vetor solucao: [010000110010010] peso 19 beneficio 29 Tempo de execucao: 5.000000 ms

Melhor solucao Aleatoria em 25000 iteracoes  
vetor solucao: [010000100011010] peso 19 beneficio 30 Tempo de execucao: 10.000000 ms

Melhor solucao Aleatoria em 50000 iteracoes  
vetor solucao: [010000100011010] peso 19 beneficio 30 Tempo de execucao: 20.000000 ms

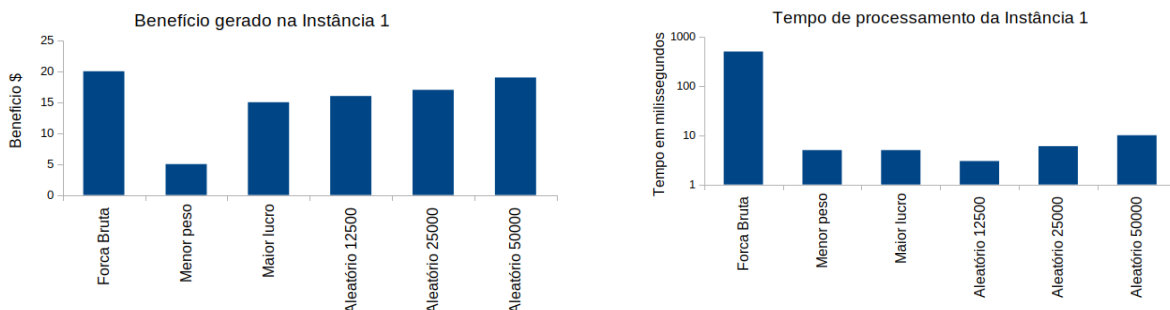
Arquivo disponível no link: [https://drive.google.com/file/d/1MumNkJM8NCV9cPcJ4nrtjv\\_WSfB1mP5y/view?usp=sharing](https://drive.google.com/file/d/1MumNkJM8NCV9cPcJ4nrtjv_WSfB1mP5y/view?usp=sharing)

### Etapa 5 - Experimentação

Uma vez as funções implementadas, o código comentado, e as saídas conferidas, pode-se então realizar a etapa de experimentação. Instâncias de experimentação são entradas usadas para avaliar o seu algoritmo, neste caso, cada instância é uma entrada diferentes, que irá gerar uma saída diferentes. Você devera gerar experimentos com as seguintes instâncias (os experimentos maiores podem demorar a serem executados, é normal). Os experimentos devem ser realizados em seu computador (ex.: Code::Blocks), e não em ambientes de rede como o *gdbonline*.


- Instância 1:
  - [https://drive.google.com/file/d/1oBohLWriAdyEWS9OMUZoSawJHJ1uK\\_Mb/view?usp=sharing](https://drive.google.com/file/d/1oBohLWriAdyEWS9OMUZoSawJHJ1uK_Mb/view?usp=sharing)
- Instância 2:
  - <https://drive.google.com/file/d/1eKOMzoWDt99rDCxg1ckChhTPbwKiyXb8/view?usp=sharing>
- Instância 3:
  - [https://drive.google.com/file/d/1EL75Jxb8opTmlbPc-LJb7x\\_iM0RpHb8d/view?usp=sharing](https://drive.google.com/file/d/1EL75Jxb8opTmlbPc-LJb7x_iM0RpHb8d/view?usp=sharing)
- Instância 4:
  - <https://drive.google.com/file/d/1n9FmCvnqlqtqhxapULkxit6L5RIEMKh/view?usp=sharing>

A cada experimento realizado com cada instância, você deve anotar os tempos de execução e benefícios gerados por cada abordagem. Utilize uma função de marcação similar a usada no TPI. Ao final, você deverá gerar gráficos como o exemplo abaixo. Sendo que tais gráficos devem fazer parte do relatório que você deve **obrigatoriamente** entregar. Exemplo de gráfico para a instância 1:




Você deve implementar o código em modulos, exemplo:

1. *main.c* – Código principal.
2. *funcoes.h* – Cabeçalhos das funções implementadas.
3. *funcoes.c* – Arquivo de implementações das funções!

 funcoes.c

 funcoes.h

 main.c

### O que entregar e pontuação!

- Código completo, testado e comentado %50 da nota (entregar por link do gdb online).
- Documentação, em **pdf**, %50 da nota, contendo:
  - Descrição do que cada componente do grupo fez, quais as maiores dificuldades e problemas encontrados;
  - Introdução, explicando o problema da mochila, e aplicações. Para isso, pesquisar aplicações da mochila binária. *Existem várias*. Pesquisem. Gastem tempo nesta parte.
  - Experimentação completa, com gráficos bem feitos, organizados e **explicados**. Descrição da máquina utiliza para realizar os experimentos. Atenção, gráficos de tempo são feitos em escala de logaritmo. Gráficos não explicados, terão pontuação descontada. Os experimentos não devem ser feitos no *onlinegdb*. Devem ser feitos gráficos para o tempo e para o benefício gerado em cada experimento.
  - Conclusões, e explicações, como: porque ao aumentar o tamanho da entrada do problema, o tempo de execução aumentou para o algoritmo de força bruta?

Permitido grupos de ate **3 pessoas, sem exceções!** Se alguém quiser fazer em dupla, ou de forma individual, conta e risco de vocês. Se você fizer o básico, mas de maneira perfeita, você garante 100% desta trabalho.

**Data de entrega: Dia da Prova II.**  
**Não serão aceitos trabalhos entregues depois do prazo.**  
**Fim de semestre, não é possível adiar a entrega.**

Gerencie seu tempo com sabedoria!  
**Em caso de plágio, será atribuído 0.**

Bom trabalho!!!