

TRABALHO PRÁTICO**Observações:**

1. Comece a fazer este trabalho imediatamente. Você nunca terá tanto tempo para resolvê-lo quanto agora!
2. **Data de entrega:** 31 de maio de 2017, até às **23:59 horas**, ou antes.
3. **Submissão:** Envie este trabalho para o endereço eletrônico `esub.para.loureiro@gmail.com` tendo como assunto `MD: "seu nome completo"` e como anexo um arquivo zip, descrito abaixo, com o nome `MD_"SeuNomeCompleto".zip` onde o string "SeuNomeCompleto" é o seu nome completo sem espaços em branco.

Exemplo para o aluno Zoroastro Felizardo e Sortudo:

- Assunto: `MD: Zoroastro Felizardo e Sortudo`
 - Arquivo zip: `MD_ZoroastroFelizardoESortudo.zip` contendo apenas os seguintes três arquivos:
 - (a) Os arquivos `analise1.c`, `analise2.c` e `analise3.c`: arquivos fontes na linguagem C, referentes aos três problemas abaixo, respectivamente.
 - (b) O arquivo `analise.pdf`: documentação
 - (c) O arquivo `leiametext`: instruções de execução
4. **Entrada:** Para cada um dos problemas abaixo, você deve fornecer dois números inteiros, de acordo com as respectivas restrições, sendo que os dois números devem ser fornecidos em uma única linha, sendo primeiro o r e depois o n .
 5. **Saída:** De acordo com a observação abaixo.
 6. **Plataforma computacional:** O seu trabalho deve ser executado em alguma máquina do ambiente computacional do Departamento de Ciência da Computação da UFMG, onde os monitores irão avaliá-lo.
 7. **Linguagem:** Você deve escrever o seu programa obrigatoriamente na linguagem de programação C.
 8. **Documentação:**
 - Uma documentação “mínima” que explique as fases de especificação, projeto e implementação, incluindo a descrição de como você resolveu cada parte deste trabalho incluindo uma discussão sobre o projeto das estruturas de dados.
 - Um arquivo `leiametext`, a ser incluído no arquivo zip, como informações sobre o ambiente computacional para executar o seu TP bem como todas as instruções necessárias.
 9. **Testes:** O seu programa será avaliado para diferentes valores de n e r .

Análise Combinatória

Para cada problema de análise combinatória descrito abaixo, você deve projetar um algoritmo para resolvê-lo. Observe os seguintes pontos:

- Você deve projetar uma solução iterativa para resolver cada problema, ou seja, não se deve projetar um algoritmo recursivo.
- Você deve imprimir cada sequência do “menor” valor para o “maior”, ao considerarmos cada sequência como se fosse um número. Assim, as sequências devem ser impressas em ordem crescente.
- Você deve fazer dois dos três problemas abaixo. Cada problema vale seis pontos. Se desejar fazer o terceiro, ele valerá quatro pontos.

1. Permutação com repetição

Projete um algoritmo para gerar sequências de comprimento r , $2 \leq r \leq 10$, sendo que cada número da sequência pode variar de 1 a n , $1 \leq n \leq 20$, conforme ilustrado abaixo.

$$\begin{array}{ccccc} \frac{1 \dots n}{1} & \frac{1 \dots n}{2} & \frac{1 \dots n}{3} & \frac{1 \dots n}{\dots} & \frac{1 \dots n}{r} \end{array}$$

Por exemplo, para $r = 2$ e $n = 3$ deve ser impresso:

1	1
1	2
1	3
2	1
2	2
2	3
3	1
3	2
3	3

2. Permutação sem repetição

Projete um algoritmo para gerar sequências de comprimento r , $2 \leq r \leq 10$, sendo que o primeiro número da sequência pode variar de 1 a n , o segundo não deve conter o anterior, o terceiro não deve conter nenhum dos dois anteriores e o r -ésimo número não deve conter nenhum dos $r - 1$ anteriores, sendo que $1 \leq n \leq 20$, conforme ilustrado abaixo (“#op” indica a quantidade de opções para aquela posição).

$$\begin{array}{ccccc} \frac{\#op = n}{1} & \frac{\#op = n-1}{2} & \frac{\#op = n-2}{3} & \frac{\dots}{\dots} & \frac{\#op = n-r+1}{r} \end{array}$$

Por exemplo, para $r = 3$ e $n = 3$ deve ser impresso:

1	2	3
1	3	2
2	1	3
2	3	1
3	1	2
3	2	1

3. Combinação com repetição

Projete um algoritmo para gerar combinações com repetição de r elementos, $2 \leq r \leq 10$, de um conjunto de n elementos formado pelos números de 1 a n , sendo que $1 \leq n \leq 5$. Lembre-se que numa combinação, a ordem não é importante.

Por exemplo, para $r = 3$ e $n = 4$, ou seja, o conjunto $\{1, 2, 3, 4\}$ deve ser impresso:

1	1	1
1	1	2
1	1	3
1	1	4
1	2	2
1	2	3
1	2	4
1	3	3
1	3	4
1	4	4
2	2	2
2	2	3
2	2	4
2	3	3
2	3	4
2	4	4
3	3	3
3	3	4
3	4	4
4	4	4