

LISTA DE EXERCÍCIOS 00

INSTRUÇÕES:

- 1.) Providenciar ambiente para o desenvolvimento de programas em C/C++:
- compilador (gcc ou MinGW) e arquivos complementares (io.h, karel.h)
 - editor de textos para programas (pode ser integrado como o JGrasp)
 - atalhos ou facilitadores para tradução (compilação) e execução.

SUGESTÃO: Ver dicas nas relações de **links** disponíveis mais abaixo.

- 2.) Testar a instalação do ambiente para o desenvolvimento de programas em C:

- executar programa para abrir uma janela de comandos do sistema:

cmd (Windows) ou terminal (Linux/Mac)

- executar na janela de comandos:

gcc -v

OBS.: Conferir se as versões, de acordo com a instalação.

- 3.) Testar a execução e a tradução (compilação) de um programa em C

- extrair os arquivos contidos em 2020-2_aed1_Exemplo_00 para a pasta associada à janela de comandos do sistema:

AED1.c

OBS.:

Para os exemplos a seguir serão consideradas as definições padrões, realizadas após a instalação da ferramenta da linguagem C:

- compilar o programa fonte (texto) em C:

no Linux : gcc -o ./AED1 ./AED1.c
no Windows: gcc -o AED1.exe AED1.c

- executar o programa objeto (executável):

no Linux : ./AED1
no Windows: AED1

- observar as saídas

- anotar dúvidas, questões e comentários, para posteriores discussões, ao final do arquivo com o programa fonte, colocando o texto entre os sinais indicados abaixo:

```
/*  
 // colocar o texto aqui <- (comentario)  
*/
```

OBS.: Evitar o uso de acentos e cedilha. Não usar nomes contendo espaços em branco.

- 4.) Para familiarizar-se com a linguagem C, editar e compilar programas para cada um dos exemplos no material anexo à apostila (Anexo_C_02), segundo o padrão sugerido no modelo Lista00.c.

SUGESTÃO: Montar um menu para facilitar a execução de testes, conforme exemplo em 2020-2_aed1_Lista_00.c .

- Cada conjunto deverá conter, em seu cabeçalho, como comentário (`/** e */`), a documentação essencial: nome e matrícula, identificação, objetivo, parâmetros e condições especiais, se houver, e relatório de testes (exemplos de valores usados e condições testadas).
- O programa contendo todos os métodos deverá ser guardado para ser usado como referência posterior, poderá ser requisitado em laboratório e ou mesmo requisitado para envio ao SGA.

SUGESTÃO: Guardar cópia também em outro local, (por exemplo, OneDrive, Google Drive etc), usando sua conta de **email** (____ @ sga.pucminas.br). Fazer controle do desenvolvimento através de versões é recomendado. Guardar cópias e não descartá-las.

- Antes da elaboração das primeiras atividades recomenda-se a leitura do arquivo contendo o roteiro para elaboração de programas (2020-2_aed1_roteiro.pdf).
- Em caso de dúvidas, solicitar esclarecimentos, o quanto antes, aos professores ou aos monitores. Os horários de disponibilidades serão informados oportunamente.

SUGESTÃO: Apresentar e discutir programas em laboratório é um excelente exercício de comunicação técnica e poderá auxiliar os professores no endereçamento de questões comuns.

Links (acessados em 30/07/2020):

Compilador para as linguagens C/C++

- 1.) Ambiente mais atual e próximo ao de laboratório
(versão 9.3.0 - 20200720)

<https://www.msys2.org>

<https://github.com/msys2/msys2-installer/releases/tag/2020-07-20>

- 2.) Ambiente mais atual e portátil para Windows 10
(versão 10.2.0 - 30/07/2020)

https://github.com/brechtsanders/winlibs_mingw/releases/download/10.2.0-7.0.0-r2/winlibs-x86_64-posix-seh-gcc-10.2.0-mingw-w64-7.0.0-r2.7z

https://github.com/brechtsanders/winlibs_mingw/releases/download/10.2.0-7.0.0-r2/winlibs-x86_64-posix-seh-gcc-10.2.0-mingw-w64-7.0.0-r2.zip

Ambiente integrado para desenvolvimento (IDE)

- 1.) Ambiente portátil para Windows 10 e Linux
(versão 2.0.6_04 - 21/05/2020)

<https://jgrasp.org>

- 2.) Ambiente portátil para Windows 10 e Linux

<https://vscodium.com>

Para textos introdutórios

- 1.)

<https://raw.githubusercontent.com/tscheffl/ThinkC/master/PDF/Think-C.pdf>

- 2.)

<http://www.greenteapress.com/thinkcpp/thinkCScpp.pdf>