

1.1 Se ainda não o fez, comece por ativar a sua conta nos computadores dos laboratórios do DCC (LabCC):

1. faça um *login* com o utilizador `password` (não necessita de palavra-chave neste passo);
2. leia a condições de uso, e responda às questões que lhe forem colocadas;
3. Introduza os seus dados pessoais;
4. Escolha a sua palavra-chave e introduza-a (em duplicado, para evitar erros).

Algumas sugestões:

- Evite palavras-chave simples: uma boa palavra-chave deve incluir letras minúsculas e maiúsculas, números e sinais de pontuação; além disso deve ter pelo menos 8 caracteres no total.
- A palavra-chave é pessoal e intransmissível.
- Pode sempre modificar a palavra-chave mais tarde usando este mesmo procedimento, por exemplo se esquecer a palavra-chave que escolheu.

Os computadores dos laboratórios correm uma sistema operativo GNU/Linux com ambiente gráfico. Comece por procurar nos menus algumas aplicações que poderão ser úteis:

Chromium/Firefox: navegadores de páginas *web*;

Adobe Acrobat Reader: um visualizador de ficheiros PDF (*Portable Document Format*);

Terminal: um interpretador de comandos ("*shell*");

GNU Emacs: um editor de texto avançado.

1.2 Usado o GNU Emacs, passe o seguinte programa com uma mensagem de boas-vindas e grave-o num ficheiro `exemplo.c`

```
main() {  
    printf("Ser ou não ser, eis a questão.\n");  
}
```

- (a) Compile o programa com o seguinte comando de "*shell*" de Linux:

```
gcc -Wall -o exemplo exemplo.c
```

- (b) A compilação produz um ficheiro executável `exemplo` no diretório corrente (e possivelmente algumas mensagens). Execute o programa com o seguinte comando de "*shell*":

```
./exemplo
```

- (c) Baseado-se no exemplo apresentado na aula teórica, corrija o programa de forma a que a compilação deixe de produzir mensagens ("*warnings*").

▷ **1.3** Considere a versão final do programa de cálculo de volume de uma caixa apresentado na aula teórica.

- (a) Use um editor de texto (e.g. GNU Emacs) para passar cuidadosamente o programa para um ficheiro `volume.c` (para se familiarizar com a sintaxe não use simplesmente “copy-paste”, componha mesmo o programa).
- (b) Compile o programa com `gcc -Wall -o volume volume.c` e corrija eventuais erros. Deve obter um ficheiro executável `volume` no diretório corrente.
- (c) Experimente correr o executável `volume` para calcular alguns volumes:

L	W	H	Volume
11	5	6	330
5	6	11	?
11	12	13	?
120	560	1700	?
1500	1500	1500	?

- (d) Submeta o seu programa ao sistema de testes automáticos:

<https://codex.dcc.fc.up.pt/cc1003>

(Use o *login* e *password* do LabCC para se autenticar.)

1.4 Considere ainda o programa da questão anterior. Para $L = W = H = 1500$, podemos constatar que o programa não calcula o volume correto (porquê?).

Qual será o lado do maior cubo para o qual o programa dá resultados corretos? Justifique a sua resposta. *Sugestão*: nas arquiteturas Intel x86 (como os PCs no laboratório) o tipo `int` representa valores apenas entre -2^{31} e $2^{31} - 1$.

1.5 Considere as declarações de variáveis.

```
int a, b, c;  
float x, y;
```

Identifique e corrija os erros nas instruções de escrita e leitura seguintes.

- (a) `printf("d%d%d\n", a, b, c);`
- (b) `printf("%d;%d\n", x, y);`
- (c) `printf("%d,%d\n", a, b, c);`
- (d) `printf("%f,%f,%d\n", a, b, x);`
- (e) `printf("%.2\n", x);`
- (f) `scanf("d", &a)`
- (g) `scanf("%f",&b);`
- (h) `scanf("%f",y)`

1.6 Modifique o programa de cálculo de volume de uma caixa para operar com dimensões de tipo `float`. Deve alterar não apenas as declarações de variáveis, mas também a leitura de dados e escrita de resultados.