

**Programação em Sistemas Computacionais**

Segunda série de exercícios – Inverno de 2013/2014

---

Os exercícios seguintes deverão ser resolvidos em *assembly* IA-32, usando a linguagem C para o código de teste. Não se esqueça de testar devidamente o código desenvolvido, bem como de o apresentar de forma cuidada, apropriadamente indentado e comentado. Não é necessário relatório. Contacte o docente se tiver dúvidas. Encoraja-se também a discussão de problemas e soluções com os seus colegas, mas salienta-se que a partilha directa de soluções leva, no mínimo, à anulação das entregas de todos os envolvidos.

1. Implemente a função `xstrlcat`, que será a sua versão da função `strlcat`, usada nos sistemas OpenBSD e FreeBSD. Veja a documentação em: <http://linux.die.net/man/3/strlcat>

```
size_t xstrlcat(char *dst, const char *src, size_t size);
```

2. Escreva a função `printout`, que usa a instrução `int $0x80`, com o valor 4 em `eax`, para invocar a função de sistema `write` de modo a que esta afixe a *string* `str` no *standard output*. Os argumentos de `write` devem ser colocados, por ordem, nos registos `ebx`, `ecx` e `edx`. Para escrever no *standard output*, utilize o valor 1 para o parâmetro `fd`. Deve invocar `strlen` para calcular o comprimento da *string* `str`. A documentação de `write` está em: <http://linux.die.net/man/2/write>

```
void printout(char * str);
```

3. Construa a função `convertAll`, que aplica a função indicada por `convertOne` a cada um dos `len` elementos do *array* `input`, indicando como espaço de destino da conversão a posição equivalente no *array* `output`. Tenha em conta que os elementos de entrada têm *idim bytes*, enquanto os de saída têm *odim*, pelo que posições equivalentes poderão corresponder a *offsets* diferentes. O argumento `ctx` é passado, sem modificação, como terceiro argumento de `convertOne`. No final, a função `convertAll` retorna o endereço da primeira posição não ocupada de `output`.

```
void * convertAll(const void * input, size_t idim, size_t len,  
                 void * output, size_t odim,  
                 void (*convertOne)(const void * in, void * out, void * ctx),  
                 void * ctx);
```

4. O programa fornecido em anexo a este enunciado pede 2 senhas de acesso ao utilizador. Pretende-se que descubra quais são as senhas correctas a introduzir, por análise do conteúdo do ficheiro fornecido e da sua execução, tendo sempre em conta que a introdução de uma senha errada tem consequências indesejáveis. Existe um programa distinto para cada aluno. Use apenas o que tem o seu número de aluno.

NOTA: Não desista à primeira contrariedade, mas não deixe de entregar os resultados a tempo só porque lhe falta uma das senhas.

Data limite de entrega: 13 de Novembro de 2013