ISEP
INSTITUTO SUPERIOR
DE ENGENHARIA DO PORTO
POLITÉCNICO DO PORTO

Departamento de Engenharia Informática
Lic. em Engenharia Informática
Arquitectura de Computadores – ARQCP
Exercício de Avaliação Individual – 2022/2023

Exercício modelo — Duração: 80 minutos – Com consulta

1. Considere a seguinte estrutura relativa à cifragem de uma mensagem de texto.

```
typedef struct{
	short *texto_cifrado;
	int tamanho_cifrado;
	char *texto_original;
}Cifra;
```

O campo **texto_cifrado** armazena o texto cifrado sob a forma de um vetor de *shorts*. O número de elementos desse vetor está armazenado em **tamanho_cifrado**. O campo **texto_original** irá armazenar a mensagem de texto original. O processo usado na cifra implica que o texto original terá sempre o dobro dos elementos de **texto_cifrado** mais um byte para o zero que deverá terminar a *string*.

Implemente em Assembly a função **void decifra_par(short numero, short chave, char *c1, char *c2)**. A função decifra dois caracteres a partir do parâmetro **numero**, escrevendo-os nos endereços indicados nos parâmetros **c1** e **c2**. Para obter os dois caracteres deve efetuar os seguintes passos:

- Aplicar um XOR do valor presente em **chave** com o *short* passado no parâmetro **numero**. Este passo dá origem a um *short* que representa um par de caracteres, pela ordem inversa à qual devem aparecer no texto original
- No byte menos significativo desse *short* está o caracter que deve ser escrito no endereço indicado em **c2**
- No byte mais significativo desse *short* está o caracter que deve ser escrito no endereço indicado em **c1**

Esta função deve ser desenvolvida no ficheiro **decifra_chars.s**

2. Implemente em Assembly a função **void decifra_string(Cifra *c, short chave)**. Esta função decifra o conteúdo do texto presente no campo **texto_cifrado** da estrutura representada pelo parâmetro **c**, armazenando o resultado no campo **texto_original**. Para tal, deverá invocar para cada elemento presente em **texto_cifrado** a função **decifra_par(...)** desenvolvida no exercício anterior. Para invocar a função **decifra_par(...)** considere que **c1** e **c2** são endereços de variáveis locais à função **decifra_string(...)**. Assuma que **texto_cifrado** tem sempre um número par de elementos.

Exemplos:

Texto cifrado	Texto original	Chave
12105, 2136, 7692, 8009, 23405, 10621, 14460	Teste de ARQCP	31532
15956, 6721, 7692, 8009, 23405, 10621, 14460	Exame de ARQCP	31532

Esta função deve ser desenvolvida no ficheiro **decifra_texto.s**

3. Implemente o ficheiro principal da aplicação (**'main.c'**) em linguagem C. Pretende-se decifrar diversas mensagens. Para tal, o programa principal deve, dado um número de mensagens a decifrar, alocar dinamicamente um vetor do tipo **Cifra**. Para efeitos do exercício assuma que pretende decifrar duas mensagens, com os seguintes dados (não necessita escrever o código para as leituras):

Texto cifrado	Chave
12105, 2136, 7692, 8009, 23405, 10621, 14460	31532
15956, 6721, 7692, 8009, 23405, 10621, 14460	31532

O seu programa deve:

- Alocar dinamicamente espaço para os dados apresentados (Note que não deve existir desperdício!);
- Preencher os campos **texto_cifrado** e **tamanho_cifrado** de cada uma das estruturas;
- Invocar a função **decifra_string(...)** de modo a decifrar as mensagens originais;
- Imprimir as mensagens originais obtidas;
- Libertar as regiões de memórias criadas na resolução do problema.

Faça uma Makefile para construir o executável na qual exista uma regra especifica para cada fase de construção (compilação; *assemblagem* e *linkagem*) e para ficheiro!

Notas importantes

- Não deve alterar o cabeçalho das funções
- Serão penalizadas as seguintes situações: Código que não compila; Código com warnings; Código não indentado; Código sem comentários adequados
- **No final deve criar um zip com todos os ficheiros da sua solução, com o nome "<turma>_<numero do aluno>.zip" e deve submete-lo usando o link que foi criado no moodle para esse efeito.**
- **Será considerada como fraude qualquer tipo de troca de dados/informação com terceiros durante a aula sem a autorização do docente.**