Assembly do IA-32 em ambiente Linux

TPC7 e Guião laboratorial

Alberto José Proença

Objectivo

A lista de exercícios/tarefas propostos no TPC7 / Guião laboratorial continua a analisar o **suporte a estruturas de controlo e a funções em C**, no IA-32, com recurso a um depurador (*debugger*). Os exercícios para serem resolvidos antes da aula TP estão assinalados com uma caixa cinza. **Não esquecer** que estes trabalhos experimentais deverão ser realizados no servidor Unix de SC, à semelhança dos trabalhos anteriores.

Ciclo *For*

1. Na diretoria `/temp` no servidor remoto Unix de SC encontra-se disponível o ficheiro executável `m-contaN` ; copie-o para a sua directoria e realize o trabalho a partir de lá. O ficheiro contém um programa executável que calcula o somatório dos dígitos numa cadeia de carateres, a partir de uma dada posição (tirando partido do facto de que o valor em hexadecimal do código ASCII do símbolo "0" é `0x30`).
Este ficheiro foi obtido após a execução do comando

```
gcc –Wall -O2 -I. contaN.c m-contaN.c -o m-contaN
```

a partir da consola de um sistema Linux.
Contudo, após a execução desse comando, o ficheiro `contaN.c` ficou danificado...
O ficheiro `m-contaN.c` contém o seguinte:

```
#include <stdio.h>
#include <stdlib.h>

int contaN(char *s, int c);

int main()
{
   char cadeia[50];
   int c;
   printf("Introduza a cadeia de carateres -->\n");
   scanf("%s",cadeia );
   printf("Qual a posicao inicial na cadeia de carateres -->\n");
   scanf("%d",&c );
   printf("O somatorio dos digitos na cadeia é -->%d\n",
          contaN(cadeia,c));
   exit(0);
}
```

a) **Teste** o funcionamento do programa a partir da consola usando como entrada de dados uma cadeia de carateres contendo alguns algarismos em decimal (ex.: "`1239aaswe67899`") e um inteiro para a posição inicial na cadeia de carateres.

b) Execute de novo o mesmo programa através do `gdb`. Use os comandos disponíveis para examinar código, de forma a visualizar o código simbólico ("desmontado" ou *disassembled*) correspondente à função (e apenas este). Escreva aqui o que obteve:

c) Anote cuidadosamente o código visualizado na alínea anterior tendo em consideração que o resultado da função é devolvido no registo `%eax`.
Identifique no código:

- os registos que são atribuídos às variáveis locais `result` (_____) e `i` (_____)
- os registos que são usados com os argumentos da função _____
- a condição de teste do ciclo `for` _____
- o modo como a variável `i` é atualizada _____
- o código decimal correspondentes aos dígitos representados em *ASCII* _____
- a expressão em C que atualiza o valor de `result` no ciclo_____

d) Com base no resultado das alíneas anteriores, recupere o ficheiro `contaN.c`. Desconfia-se que a estrutura da função que estava em `contaN.c` era do tipo:

```
int i;
int result;
???
for ( ??? ; s[i]!= ??? ; ???)
   if (s[i] >= '0' && ??? )
      result += ??? ;
return result;
```

(*Para fazer depois da sessão laboratorial*)