

Guião de Laboratório de Arquitectura de Computadores

Simulação 5.14 – Listas ligadas

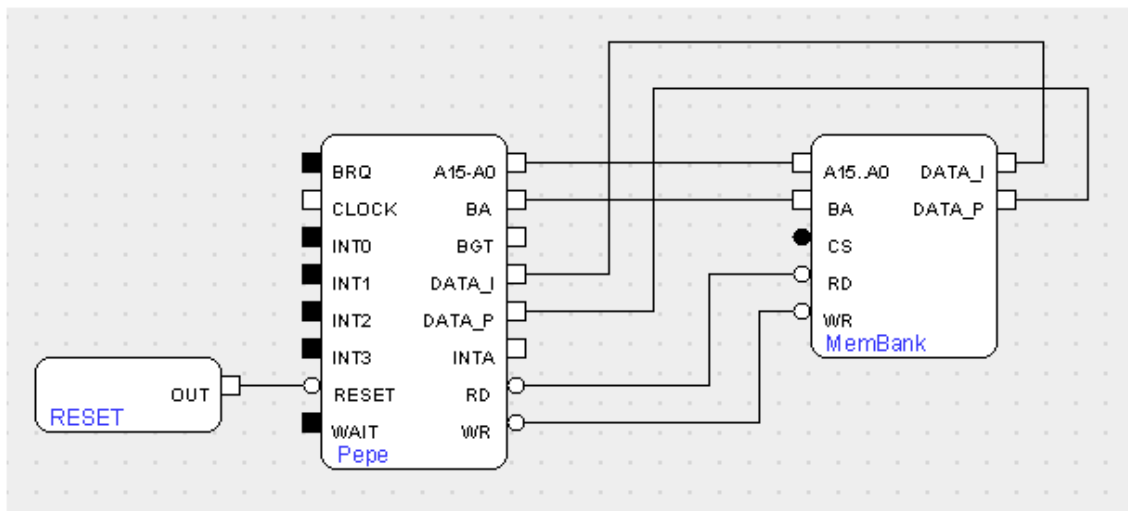
1 – Objectivos

Esta simulação ilustra o funcionamento das listas ligadas, tomando o Programa 5.15 como base. Os aspectos cobertos incluem os seguintes:

- Disposição das fichas em memória e seu conteúdo, incluindo apontadores;
- Funcionalidades básicas das listas (acesso, travessia, inserção e remoção);
- Evolução dos valores dos registos relevantes.

2 – Circuito

O ficheiro “pepe.cmod” implementa o circuito da Fig. 4.7. A simulação 4.1 contém indicações mais detalhadas sobre a sua utilização no simulador.



3 – Simulação do programa 5.15

Carregue este circuito no simulador e passe para Simulação.

Abra o painel do PEPE e compile e carregue (📁) o ficheiro “programa5-15.asm”.

Este programa ilustra a manipulação de listas ligadas em memória. Confira a explicação sobre o seu funcionamento no livro.

Cada elemento da lista foi declarado por directiva WORD, posicionadas manualmente em memória com PLACES. Note os apontadores, em cada elemento, para o elemento seguinte. É esta referência em cadeia que lhe dá o nome de lista ligada.

A zona de dados na memória terá o seguinte aspecto:

Addr...	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1190	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11A0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11B0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11C0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11D0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11E0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11F0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1200	0	0	0	0	12	14	B	D5	0	CF	0	22	4	E2	0	0
1210	0	0	0	0	12	26	7	BA	0	E1	0	17	3	DE	0	0
1220	0	0	0	0	0	0	0	0	9	55	0	D9	0	2F	6	72
1230	0	0	0	0	0	0	4	3	0	EA	0	3A	7	E9	0	0
1240	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1250	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1260	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1270	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1280	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1290	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12A0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Verifique o apontador, no início de cada elemento, para o elemento seguinte. Note que a sequência de endereços de elementos é agora

1204H -> 1214H -> 1226H -> null

Inicialmente, a lista tem 3 elementos, que começam nos endereços 1204H, 1214H e 1226H. A lista acaba neste último pela simples razão de que o seu apontador está a null (0000H). O primeiro elemento designa-se por cabeça da lista, e o último por cauda.

Execute o programa passo a passo.

A rotina Idade ilustra a travessia da lista, elemento a elemento, fazendo processamento sobre cada um deles. Percorre os elementos todos, desde a cabeça até à cauda ou até achar o elemento que pretende. Tem de haver uma apontador conhecido para a cabeça da lista, neste caso "ficha1". A partir daí, é só obter em cada elemento o apontador para o elemento seguinte.

Neste exemplo, em cada elemento testa o número mecanográfico do empregado, passando ao próximo elemento se este não tiver esse número, no campo apropriado. Quando descobrir o elemento pretendido, obtém a idade e regressa. Vá vendo a evolução dos registos.

As rotinas Insere e Remove ilustram na prática o explicado na Fig. 5.12. verifique este comportamento, em execução passo a passo. Para comodidade, o elemento a inserir já está criado, nas declarações de dados iniciais.

A memória após a execução da rotina Insere será:

Addr...	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1190	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11A0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11B0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11C0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11D0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11E0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11F0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1200	0	0	0	0	12	14	B	D5	0	CF	0	22	4	E2	0	0
1210	0	0	0	0	12	34	7	BA	0	E1	0	17	3	DE	0	0
1220	0	0	0	0	0	0	0	0	9	55	0	D9	0	2F	6	72
1230	0	0	0	0	12	26	4	3	0	EA	0	3A	7	E9	0	0
1240	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1250	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1260	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1270	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Note que a sequência de endereços de elementos é agora

1204H -> 1214H -> 1234H -> 1226H -> null

Pode também observar-se que numa lista ligada os endereços dos elementos não têm de estar sempre a crescer, ou decrescer. Tudo depende do endereço do elemento que se insere e do ponto da lista em que se insere.

Após a execução da última rotina, Remove, o aspecto da memória será:

Addr...	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1190	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11A0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11B0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11C0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11D0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11E0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11F0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1200	0	0	0	0	12	34	B	D5	0	CF	0	22	4	E2	0	0
1210	0	0	0	0	12	34	7	BA	0	E1	0	17	3	DE	0	0
1220	0	0	0	0	0	0	0	0	9	55	0	D9	0	2F	6	72
1230	0	0	0	0	12	26	4	3	0	EA	0	3A	7	E9	0	0
1240	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1250	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1260	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

A sequência de endereços de elementos é agora

1204H -> 1234H -> 1226H -> null

O elemento 1214H foi removido. Ele até continua lá, mas já não está acessível a partir da cabeça pela cadeia de apontadores.