Universidade Positivo Sistemas de Informação
11/02/2019 - 12:34:01.4
Estruturas de Dados e Arquivos Prof Dr P Kantek (pkantek@up.edu.br)
Exercício prático de recursividade VIVO251a V: 3.18
1
Exercício : 1 _____ / _____ / _____

Recursividade

A recursividade é uma técnica preciosa na programação. Derivada do conceito matemático de indução, em algumas linguagens de programação (por exemplo LISP) ela é senão a única, pelo menos a principal estrutura de repetição. Lá, não são usados nem *while*, nem *repeat*, nem *for*, apenas recursões.

Veja um exemplo jocoso, mas bem real de recursividade:

Em programação, uma função recursiva tem sempre 3 características imprescindíveis, a saber

- Ela chama a si própria
- Existe pelo menos um caso básico, no qual não há chamada recursiva e que ao final, garante que a função não entre em loop eterno
- A cada chamada, o universo da função é menor (sempre em direção ao caso básico)

A seguir, 3 funções recursivas são apresentadas, pedindo-se que o aluno ache o resultado solicitado.

Sejam dados os seguintes algoritmos

1: função A (inteiro X) : inteiro
2: AUX : inteiro
3: imprima 'A'
4: se $(X > 4)$
5: $AUX \leftarrow A(X-3)$
6: se $(X \geq 10)$
7: devolva $B(X \times 2) + AUX$
8: senão
9: devolva $B((X \times 2) + 1) + AUX$
10: fim{se}
11: senão
12: devolva $(Z + C(X))$
13: fim{se}

1: função B (inteiro W) : inteiro
2: imprima 'B'
3: se $(W < 5))$
4: devolva (W + Y)
5: senão
6: devolva (2+ B(W-3))
7: fim{se}

1: função C (inteiro K) : inteiro
2: imprima 'C'
3: se $(K > 2)$
4: devolva 1+C(K-1)
5: senão
6: devolva 1
7: fim{se}

Se forem dadas variaveis globais $Y = 9$ e $Z = 10$, qual seria o valor de $A(9)$?

a O exercício começa com A(9) e a função A é chamada imprimindo um 'A' Como $9 > 4$ (linha 4 de A), deve-se calcular AUX que vem a ser A(9-3) ou A(6). Fica parada na memória uma instância da função A, na linha 5, com X=9.

b A seguir, calcula-se A(6), e a função imprime um 'A' (impressão AA). Como $6 > 4$ (linha 4 de A), calcula-se AUX, agora valendo A(6-3) ou A(3). Fica parada na memória uma instância da função A, na linha 5, com X=6.

c A seguir calcula-se A(3) e a função imprime um 'A' (impressão AAA) Como 3 não é > 4, (linha 4 de A) esta função quer devolver 10+C(3). Fica parada na memória uma instância da função A, na linha 12, com X=3.

d Calcula-se C(3) e a função imprime um 'C' (impressão AAAC) Como $3 > 2$, (linha 3 de C) esta função devolve 1+C(2). Fica parada na memória uma instância da função C, na linha 4, com K=2.

e Calcula-se C(2) e a função imprime um 'C' (impressão AAACC) Como 2 não é > 2 (linha 3 de C), esta função devolve 1. Logo C(2)=1.

f Retorna-se a [d], com o valor 1 e C(3)=2.

g Retorna-se a [c] e a função A(3) devolve 12. Logo A(3)=12 e AUX neste caso vale 12.

h Retorna-se a [b] e como X=6 e 6 não é maior ou igual a 10 (linha 6 de A), vai-se calcular B(13). A função B imprime um 'B' (impressao AAACCB). Como 13 não é < 5 (linha 3 de B), esta função quer devolver 2+B(10). Fica parada na memória uma instância da função B, na linha 6, com W=10.

i Calcula-se B(10). A função B imprime um 'B' (impressão AAACCBB) Como 10 não é < 5 (linha 3 de B), esta função devolve 2 + B(7). Fica parada na memória uma instância da função B, na linha 6, com W=7.

j Calcula-se B(7). Imprime-se um 'B' (impressão AAACCBBB) Como 7 não é < 5 (linha 3 de B), esta função devolve 2 + B(4). Fica parada na memória uma instância da função B, na linha 6, com W=4.

k Calcula-se B(4). Imprime-se um 'B' (impressão AAACCBBBB) Como 4 é < 5, (linha 3 de B), esta função devolve 4 + Y = 4+9=13, e caiu-se no caso básico de B.

l Retorna-se a [j] e B(7)=2+13=15.

m Retorna-se a [i] e B(10)=2+15=17.

n Retorna-se a [h] e B(13)=2+17=19.

o Retorna-se a [b] e Tem-se B(13)=19 e AUX=12, Logo A(6)=19+12=31

p Retorna-se a [a] e vai-se calcular B(19). A função imprime um 'B' (impressão AAACCBBBBB) e como 19 não é < 5, esta função devolve 2+B(16).

q Calcula-se B(16). A função imprime um 'B' (impressão AAACCBBBBBB) e como 16 não é < 5, devolve 2+B(13). Agora aqui entra uma esperteza. Não há necessidade de recalcular tudo, pois já se sabe que B(13)=19. Só precisamos ter cuidado com as impressões. Haverá um novo 'B' nas chamadas de B(13), B(10), B(7) e B(4), portanto quatro Bs a mais e a impressão final fica: AAACCBBBBBBBBBB.

r O número final é: B(16)=2+19=21. B(19)=23 e A(9)=B(19)+A(6) ou A(9)=23+31 = 54

s A resposta é A(9)=54 e a impressão é: AAACCBBBBBBBBBB

☞ Para você fazer

Note que o corpo das funções A, B e C varia de folha para folha.

1: função A (inteiro X) : inteiro
2: AUX : inteiro
3: imprima 'A'
4: se $(X > 4)$
5: $AUX \leftarrow A(X - 3)$
6: se $((X \geq 10)$
7: devolva $(B(X \times 2)) + AUX$
8: senão
9: devolva $(B((X \times 2) + 1)) + AUX$
10: fim{se}
11: senão
12: devolva $(Z + C(X))$
13: fim{se}

1: função B (inteiro W) : inteiro
2: imprima 'B'
3: se $(W < 5))$
4: devolva (W + Y)
5: senão
6: devolva (2+ B(W- 3))
7: fim{se}

1: função C (inteiro K) : inteiro
2: imprima 'C'
3: se $(K > 2)$
4: devolva 1+C(K- 1)
5: senão
6: devolva 1
7: fim{se}

Suponha Y = 11 e Z = 9
Calcule A(10) = _____ e indique a seqüência de letras a serem impressas: _____ .

Por favor, ajude-me a corrigir esta folha. Indique ao lado da mensagem impressa, quantas são as letras "B" no resultado.

Exemplo em C

Este é o exemplo. Note que a sua instância é ligeiramente diferente.

```
 1: #include<stdlib.h>
 2: #include<stdio.h>
 3: int Y=9;
 4: int Z=10;
 5: int C(int K){
 6:   printf("%c",'C');
 7:   if (K>2) {
 8:     return(1+C(K-1));
 9:   }
10:   else {
11:      return 1;
12:   }
13: }
14: int B(int W){
15:   printf("%c",'B');
16:   if (W<5) {
17:     return (W+Y);
18:   }
19:   else {
20:     return (2+B(W-3));
21:   }
22: }
23: int A(int X) {
24:   int AUX;
25:   printf("%c",'A');
26:   if (X>4) {
27:     AUX=A(X-3);
28:     if (X>=10) {
29:       return (B(X*2)+AUX);
30:     }
31:     else {
32:       return (B((X*2)+1)+AUX);
33:     }
34:   }
35:   else {
36:     return (Z+C(X));
37:   }
38: }
39: main() {
40:   printf("-A funcao A(9)=%i",A(9));
41: }
AAACCBBBBBBBBBB-A funcao A(9)=54
```