Exercício – Estruturas não contíguas

Profa. Simone Aires – Prof. Saulo

Para todas as questões, considere uma memória cujo estado é representado abaixo e as seguintes suposições:

• Suposições do exercício:

– Somente alocações do código ocupam espaço em memória RAM *na sequencia em que são executadas*;

– Arquitetura sem alinhamento mas tamanho de variáveis ponteiros não fornecido;

– Células preenchidas com **b** indicam espaço usado por outros processos;

• Mapa da memória assumido no exercício:

$Endereço_{10}$	$Endereço_{16}$	Conteúdo							
00	0x0	b_{00}	b_{01}	b_{02}	b_{03}	b_{04}	b_{05}	b_{06}	b_{07}
01	0x1								
02	0x2	b_{16}	b_{17}	b_{18}	b_{19}	b_{20}	b_{21}	b_{22}	b_{23}
03	0x3								
04	0x4								
05	0x5	b_{40}	b_{41}	b_{42}	b_{43}	b_{44}	b_{45}	b_{46}	b_{47}
06	0x6								
07	0x7								
08	0x8	b_{64}	b_{65}	b_{66}	b_{67}	b_{68}	b_{69}	b_{70}	b_{71}
09	0x9	b_{72}	b_{73}	b_{74}	b_{75}	b_{76}	b_{77}	b_{78}	b_{79}
10	0xa	b_{80}	b_{81}	b_{82}	b_{83}	b_{84}	b_{85}	b_{86}	b_{87}
11	0xb	b_{88}	b_{89}	b_{90}	b_{91}	b_{92}	b_{93}	b_{94}	b_{95}
12	0xc								
13	0xd								
14	0xe	b_{112}	b_{113}	b_{114}	b_{115}	b_{116}	b_{117}	b_{118}	b_{119}
15	0xf								

1. Calcule o tamanho mínimo de variáveis do tipo ponteiro para endereçar a memória RAM. Lembre que a unidade mínima de qualquer sistema de endereçamento é 1 Byte. Apresente seus cálculos.

Dica: Assumindo 16 Bytes de RAM ponteiros ocupam $\log_2(16)$ bits

2. Observe o código abaixo e responda as questões.

```
00  #include <stdlib.h>
01 void main()
02  {
03  char *v = (char *)malloc(sizeof(char)*3);
04  v[0]='a';
05  v[1]='b';
06  v[2]='c';
07  }
```

(a) Quanto de espaço de alocação o programa solicita?

(b) O programa tem toda a sua demanda de espaço atendida? O que ocorrerá durante a execução do programa?

(c) (**Desafio**!) Caso o programador tenha sua demanda de espaço atendida informe o endereço e o conteúdo dos bytes da RAM que serão alterados. Do contrário, apresente um código capaz de armazenar os caracteres desejados. Seu código deve realizar somente uma alocação estática. Realize quantas alocações dinâmicas desejar. Dica: use *struct* auto-referenciada.