

## EXERCICIOS HASHING

Considere a estrutura de armazenamento de dados a seguir, na resolução dos exercícios 1 e 2.

```
# define Max _____
struct Aluno{
    char nPUC[12];
    char nome[80];
    char email[30];
    char fone[12];
};
Aluno tabela[Max];
```

- 1) Supondo que o número de matrícula de um aluno da PUCSP seja composto por 8 dígitos, qual o espaço de memória (em bytes) necessário para o armazenamento do array tabela, para todos os possíveis números de matrícula? Assinalar a alternativa correta e **justificar a resposta na folha**.
  - a) 134 bytes
  - b)  $\approx 100$  Mbyte
  - c)  $\approx 1$  Gbyte
  - d)  $\approx 10$  Gbyte
  - e)  $\approx 100$  Gbyte

- 2) Se o cadastro de alunos tem não mais do que 10000 alunos, qual o espaço de memória (em bytes) necessário para o armazenamento do array tabela? **Justificar a resposta na folha pautada**.

Resposta: \_\_\_\_\_

- 3) Para cada função hash definida a seguir, determine os endereços obtidos para os valores das chaves (nesta ordem) 44, 121, 51, 68, 46 e complete a tabela de espalhamento:

0	1	2	3	4	5	6	7	8	9	10

```
int h(TipoChave c, int i){
    int k;
    k = (h1(c) + i*h2(c))%11;
    return k;
}
```

```
int h1(TipoChave c){
    int k;
    k = (2*c + 3) % 13;
    return k;
}
```

```
int h2(TipoChave c){
    int k;
    k = (c % 11) + 1;
    return k;
}
```

- 4) Considere a estrutura de armazenamento de dados de uma tabela Hashing dada a seguir e preencha com os dados referentes à inserção das chaves 71, 44, 60, 46, 49, 51 nesta ordem. Considere a função hash  $h(k) = (k \text{ MOD } 7) + 1$  e o tratamento de colisões por encadeamento com tabela de descritores.

	chave	next	ant
1			
2			
3			
4			
5			
6			
7			

	prim	ult
1		
2		
3		
4		
5		
6		
7		