



Instituto Federal de Educação, Ciência e Tecnologia de Brasília – Câmpus Taguatinga
Ciência da Computação – Estrutura de Dados e Algoritmos
Prova I – 2º/2018 – Algoritmos, Ponteiros e Ordenação
Prof. Daniel Saad Nogueira Nunes

Aluno: _____

Matrícula: _____

Data: 13 de setembro de 2018

Duração da prova: 100 minutos

Tabela de notas (uso exclusivo do professor)

Questão	Pontos	Nota
1	3	
2	3	
3	3	
4	3	
Total	12	

Observações

- Esta prova tem o total de 2 páginas (incluindo a capa) e 4 questões.
- O número total de pontos é 12.
- Certifique-se de assinar todas as folhas de resposta bem como a capa da prova.
- Leia atentamente todas as questões da prova. A interpretação do problema é crucial para o desenvolvimento correto da resposta.
- Resoluções sem justificativa não serão consideradas.
- É vedado o uso de equipamentos eletrônicos, como celulares, notebooks entre outros.
- A prova será **anulada** e medidas disciplinares serão tomadas para os alunos que “colarem” durante a avaliação.

★ Certifique-se de assinar todas as folhas de resposta.

Questão 1 (3 pontos)

Implemente o método de ordenação por inserção. Este método deverá seguir a seguinte assinatura:

```
void insertion_sort(int* v, size_t tam);
```

Questão 2 (3 pontos)

Dado dois vetores de inteiro ordenados, v_1 e v_2 , juntamente com o tamanho de cada, crie uma função que realiza a junção (*merge*) destes dois vetores em um terceiro vetor, retornando o último.

Sua função deverá possuir a seguinte assinatura:

```
int* merge(int* v1, size_t tam1, int* v2, size_t tam2);
```

Questão 3 (3 pontos)

A ordenação por classificação sobre vetores inteiros contendo valores na faixa $[a, b]$ funciona da seguinte maneira:

- Crie um vetor c de tamanho $[b - a + 1]$.
- Defina $c[i - a]$ como o número de vezes que o número i aparece no vetor v .
- Utilize c para ordenar o vetor v ao inspecioná-lo da esquerda para direita e inserir elementos em v enquanto $c[i]$ é positivo. A cada elemento de valor i inserido, $c[i]$ é decrementado.

Implemente um algoritmo utilizando a descrição acima. O seu algoritmo deverá possuir a seguinte assinatura:

```
void bucket_sort(int* v, size_t tam);
```

Questão 4 (3 pontos)

De acordo com a busca binária:

(a) ($1\frac{1}{2}$ pontos) Ilustre passo a passo este procedimento para a seguinte entrada:

- $v = \{1, 3, 5, 7, 8, 11, 13, 15, 17, 19, 21, 23\}$;
- $chave = 18$;

(b) ($1\frac{1}{2}$ pontos) Implemente a busca binária sobre um vetor de inteiros para encontrar uma determinada chave. A busca deverá retornar -1 caso o elemento não esteja no vetor. Sua implementação deverá seguir a seguinte assinatura:

```
int busca_binaria(int* v, size_t tam, int chave);
```