

Algoritmos e Complexidade

LEI (2º ano)

Perguntas do 2º Mini-Teste

Ano Lectivo de 2011/12

1. Responda às seguintes questões.

(a) Implemente a seguinte função em C.

```
int *takeWhile(int A[], int N, int p, int *l);
```

Dado um array A , ocupado nas posições 0 a $N-1$, e um inteiro p , esta função deve retornar o endereço de um novo array. O array retornado deverá conter os elementos de A que antecedem a primeira ocorrência de um valor superior a p . O número de posições válidas do array retornado deve ser devolvido na variável l .

(b) Escreva equações $T(N)$ que descrevam o tempo de execução do algoritmo que implementou, no melhor caso e no pior caso.

(c) Descreva o comportamento do algoritmo utilizando a notação assintótica Θ , O e Ω .

2. Responda às seguintes questões.

(a) Implemente a seguinte função em C.

```
int *under(int A[], int N, int p, int *l);
```

Dado um array A , ocupado nas posições 0 a $N-1$, e um inteiro p , esta função deve retornar o endereço de um novo array. O array retornado deverá conter os elementos de A que são inferiores a p . O número de posições válidas do array retornado deve ser devolvido na variável l .

(b) Escreva equações $T(N)$ que descrevam o tempo de execução do algoritmo que implementou, no melhor caso e no pior caso.

(c) Descreva o comportamento do algoritmo utilizando a notação assintótica Θ , O e Ω .

3. Responda às seguintes questões.

(a) Implemente a seguinte função em C.

```
int *dropWhile(int A[], int N, int p, int *l);
```

Dado um array A , ocupado nas posições 0 a $N-1$, e um inteiro p , esta função deve retornar o endereço de um novo array. O array retornado deverá conter os elementos de A , excluindo aqueles que antecedem a primeira ocorrência de um valor superior ou igual a p . O número de posições válidas do array retornado deve ser devolvido na variável l .

(b) Escreva equações $T(N)$ que descrevam o tempo de execução do algoritmo que implementou, no melhor caso e no pior caso.

(c) Descreva o comportamento do algoritmo utilizando a notação assintótica Θ , O e Ω .

4. Responda às seguintes questões.

(a) Implemente a seguinte função em C.

```
void topTwo(int A[], int N, int *l1, int *l2);
```

Dado um array A , ocupado nas posições 0 a $N-1$, esta função deve retornar as posições dos dois elementos de maior valor. Essas posições devem ser devolvidas nas variáveis $l1$ e $l2$ respectivamente. Assuma que $N \geq 2$.

(b) Escreva equações $T(N)$ que descrevam o tempo de execução do algoritmo que implementou, no melhor caso e no pior caso.

(c) Descreva o comportamento do algoritmo utilizando a notação assintótica Θ , O e Ω .

5. Responda às seguintes questões.

(a) Implemente a seguinte função em C.

```
typedef struct node {
    int a;
    struct node *next;
} Node, *Lista;

int *drop(Lista list, int p, int *l);
```

Dada uma lista ligada *list*, e um inteiro *p*, esta função deve retornar o endereço de um novo array. O array retornado deverá conter os elementos da lista que sucedem a posição *p*. O número de posições válidas do array retornado deve ser devolvido na variável *l*. Considere que o primeiro nó da lista está na posição 1.

(b) Escreva equações $T(N)$, sendo N o tamanho da lista, que descrevam o tempo de execução do algoritmo que implementou, no melhor caso e no pior caso.

(c) Descreva o comportamento do algoritmo utilizando a notação assintótica Θ , O e Ω .

6. Responda às seguintes questões.

(a) Implemente a seguinte função em C.

```
typedef struct node {
    int a;
    struct node *next;
} Node, *Lista;

int *dropWhile(Lista list, int p, int *l);
```

Dada uma lista ligada *list*, e um inteiro *p*, esta função deve retornar o endereço de um novo array. O array retornado deverá conter os elementos da lista, excluindo aqueles que antecedem a primeira ocorrência de um valor superior ou igual a *p*. O número de posições válidas do array retornado deve ser devolvido na variável *l*.

(b) Escreva equações $T(N)$, sendo N o tamanho da lista, que descrevam o tempo de execução do algoritmo que implementou, no melhor caso e no pior caso.

(c) Descreva o comportamento do algoritmo utilizando a notação assintótica Θ , O e Ω .