Algoritmos e Complexidade — Universidade do Minho
Teste 30 de Outubro de 2023

Número: _____ Nome: _____

1. Especificação (4 valores)

Considere a seguinte função que, dado um número inteiro x entre 1 e 9, e um vector de números com valor igualmente entre 1 e 9, procura a **primeira posição** do vector que contém o valor 10-x. Escreva uma **pré-condição** e uma **pós-condição** que descrevam este comportamento.
Observe que a pós-condição deve referir duas situações distintas, correspondentes às duas situações em que o ciclo pode terminar.

(Consider the following function which, given an integer x between 1 and 9, and a vector of numbers with a value also between 1 and 9, computes the **first position** in the vector that contains the value 10-x. Write a **precondition** and a **postcondition** that describe this behavior.
Note that the post-condition must refer to two different cases, corresponding to the two cases in which the cycle can end.)

```
int addsTo10 (int x, int a[], int N) {
     // Pre: . . .
     int i = 0 ;
     while (i<N && a[i]+x != 10)
          i=i+1 ;
     // Pos: (i==N && . . .) || (i<N && . . .)
     return i ;
}
```

Algoritmos e Complexidade — Universidade do Minho
Teste 30 de Outubro de 2023

Número: _____ Nome: _____

2. Correção (4 valores)

Escreva um invariante para a função da Questão 1. Deverá incluir o seguinte:

a. Informação sobre os valores possíveis da variável `i`
b. Informação sobre a relação entre os valores contidos nas posições já visitadas do array e o valor da variável x

(Write an invariant for the function in addsTo10. It should include (1) information about the possible values of the variable i, and (2) information about the relationship between the values contained in the positions already visited and the value of the variable x.)

Algoritmos e Complexidade — Universidade do Minho
Teste 30 de Outubro de 2023

Número: _____ Nome: _____

3. Complexidade algoritmos iterativos (8 valores)

Analise o **melhor caso**, **pior caso**, e **caso médio** da complexidade da função `addsTo10`, contando o número de comparações (`a[i]+x != 10`).
Para o caso médio apresente a solução usando somatórios.

(Analyze the best case, worst case and average case complexity of the `addsTo10` function by counting the number of comparisons (`a[i]+x != 10`). For the average case you leave the sums uncalculated.)

Algoritmos e Complexidade — Universidade do Minho
Teste 30 de Outubro de 2023

Número: _____ Nome: _____

4. Complexidade de algoritmos recursivos (4 valores)

Considere agora a seguinte função que decide se um vector de inteiros, com valores entre 1 e 9, contém ou não um par de elementos cuja soma vale 10.

(Consider the following function which tests whether or not a vector of integers, with values between 1 and 9, contains a pair of elements whose sum is 10.)

```
int par10Rec (int u[], int N) {
     int p ;
     if (N<2) return 0 ;
     p = addsTo10(u[0], u+1, N-1) ;
     if (p<N-1) return 1 ;
     return (par10Rec (u+1, N-1)) ;
}
```

De forma a analisar a complexidade desta função, identifique o melhor e pior casos e, **para o pior caso**, apresente e resolva a recorrência que traduz essa complexidade.

(In order to analyze the complexity of this function, identify the best and worst cases and, for the **worst case**, write down and solve a recurrence relation that translates this complexity)