

3. a)

```
func1 (int n, float A[], float x) {
    int k;
    float y = 0.0;
    for (x=n; k>=0; k--) {
        y = A[k] + y + x;
    }
    return y;
}
```

Custo	Nº vezes
C ₁	1
C ₂	1
C ₃	1
C ₄	n+1
C ₅	1
C ₆	1

Função de complexidade

$$f(n) = \cancel{C_1} + \cancel{C_2} + \cancel{C_3} + C_4(n+1) + \cancel{C_5} + \cancel{C_6}$$

$$f(n) = C_4(n+1)$$

Análise assintótica $\rightarrow O(n)$

b)

```
int func2 (int n) {
    int i, j, x, soma = 0;
    for (i=0; i<=n; i++) {
        for (j=0; j<n; j++) {
            for (x=0; x<n; x++) {
                soma += n;
            }
        }
    }
    return soma;
}
```

Custo	Nº vezes
C ₁	1
C ₂	1
C ₃	n+1
C ₄	$n \times (n+1) =$
C ₅	$n \cdot (n^2+1) =$
C ₆	1
C ₇	1

Função de complexidade

$$f(n) = \cancel{C_1} + \cancel{C_2} + \cancel{C_3}(n+1) + C_4 n \times (n+1) + C_5 n(n^2+1) + \cancel{C_6} + \cancel{C_7}$$

$$f(n) = C_5 n \times (n^2+1)$$

Análise assintótica $\rightarrow O(n)^3$

```

c) void func3(int *A, int n) {
    int i, j, aux;
    for (j = 2; j <= n; j++) {
        aux = A[j];
        i = j - 1;
        while (i > 0 && A[i] > aux) {
            A[i+1] = A[i];
            i = i - 1;
        }
        A[i+1] = aux;
    }
}

```

Custo	Nº vezes
C_1	1
C_2	1
C_3	$n-1$
C_4	1
C_5	1
C_6	$n(n-1)$
C_7	1
C_8	1
C_9	1

Função de complexidade:

$$f(n) = \cancel{C_1} + \cancel{C_2} + C_3(n-1) + \cancel{C_4} + \cancel{C_5} + C_6 n(n-1) + \cancel{C_7} + \cancel{C_8} + \cancel{C_9}$$

$$C_6 n(n-1)$$

caso médio
 pior + melhor / 2

pior caso: $f(n) = n(n-1)$

↳ análise assintótica: $O(n)^2$

melhor caso: $f(n) = n-1$

↳ análise assintótica: $O(n)$

caso médio: $f(n) = \frac{n(n-1) + n-1}{2}$

↳ análise assintótica: $O(n)^2$