



Algorítmica
Grado en Ingeniería Informática

Prácticas: Ejercicio de entrega

1. (9 puntos) El siguiente código compara dos cadenas s y t entre sus posiciones i y j , y devuelve una medida de distancia entre ambas cadenas. Analizar la eficiencia del siguiente algoritmo y calcular su orden en el caso peor, asumiendo que las funciones *match* e *indel* se encuentran en orden $O(1)$, y que *MATCH*, *INSERT*, *DELETE* son constantes predefinidas globales.

```
int string_compare(char *s, char *t, int i, int j)
{
    /* Note: We use odd index conventions here, see the top of page 284 */
    int k;                /* counter */
    int opt[3];           /* cost of the three options */
    int lowest_cost;      /* lowest cost */

    if (i == 0) return(j * indel(' '));
    if (j == 0) return(i * indel(' '));

    opt[MATCH] = string_compare(s,t,i-1,j-1) + match(s[i],t[j]);
    opt[INSERT] = string_compare(s,t,i,j-1) + indel(t[j]);
    opt[DELETE] = string_compare(s,t,i-1,j) + indel(s[i]);

    lowest_cost = opt[MATCH];
    for (k=INSERT; k<=DELETE; k++)
        if (opt[k] < lowest_cost) lowest_cost = opt[k];

    return( lowest_cost );
}
```

2. (1 punto) Justificar si el algoritmo tiene orden exacto o no.



Criterios de evaluación:

1. Determinación de la función T de tiempos del algoritmo: 4 puntos. Se valorará:
 1. Tipo de algoritmo: iterativo o recursivo.
 2. Análisis de las operaciones del código.
 3. Corrección de la función de tiempos.
2. Determinación del orden de la función de tiempos: 5 puntos. Se valorará la descripción detallada de todos los pasos llevados a cabo para el cálculo del orden.
3. Se valora que la justificación del ejercicio 2 es rigurosa, haciendo uso de las propiedades de los órdenes de eficiencia.