

ALGORITMOS Y ESTRUCTURAS DE DATOS II

GRADO INGENIERÍA INFORMÁTICA, CURSO 24/25

PRÁCTICA DE DIVIDE Y VENCERÁS Y ANÁLISIS DE ALGORITMOS

A. Contexto

Las **técnicas generales de diseño de algoritmos** ofrecen un amplio y variado conjunto de herramientas que se pueden usar en la resolución de muchos tipos de problemas. La aplicación de una técnica concreta debe entenderse como un proceso metódico, que empieza con la interpretación y el modelado del problema desde la perspectiva de esa técnica, sigue con la definición de las partes genéricas del esquema algorítmico (tipos de datos y funciones básicas), y acaba con la implementación, prueba, refinamiento y optimización del algoritmo. En esta práctica se aplicará la técnica divide y vencerás a la resolución de problemas de recorrido de cadenas.

B. Enunciado de la actividad

Esta actividad se hará **en grupos de dos alumnos/as**. Cada grupo tiene asignado un problema distinto, según el DNI de los componentes, X e Y (quitando la letra final o la inicial, según el caso). Los problemas asignados dependen del resultado del siguiente cálculo: $R = ((X+Y) \bmod 10) + 1$, donde R indica el número de problema a resolver.

La práctica consistirá en resolver y estudiar el problema asignado utilizando un divide y vencerás recursivo según el esquema visto en clase para división del problema en **dos** subproblemas, documentándolo en una memoria que siga el esquema indicado en el apartado C. Es condición necesaria programar el algoritmo propuesto en C++ y que funcione correctamente con los casos de prueba que pase el profesor en la revisión de la práctica; pero no es condición suficiente: habrá que tener en cuenta otras cuestiones, como eficiencia, análisis teórico y experimental, documentación...

C. Memoria de la actividad

La memoria entregada deberá contener en la portada el profesor de prácticas correspondiente, el nombre de los alumnos, grupo, subgrupo y e-mail.

Para el problema asignado al grupo, se deberán incluir los siguientes apartados (**hay que obtener al menos un tercio de la nota en cada apartado**):

1. (hasta **2** puntos) **Diseño** de una solución divide y vencerás, incluyendo pseudocódigo y explicación del algoritmo, justificando las decisiones de diseño, las estructuras de datos y las funciones básicas del esquema algorítmico.
2. (hasta **2** puntos) **Análisis teórico** del tiempo de ejecución, en los casos mejor y peor (t_m y t_M), y del orden de ejecución del algoritmo obtenido.
3. (hasta **2** puntos) **Implementación**: Programación del algoritmo (lo normal es hacer el programa tras haber diseñado y estudiado teóricamente el algoritmo). El código fuente debe ir documentado, con explicación de qué es cada variable, qué realiza cada función y su correspondencia con las funciones básicas del esquema algorítmico correspondiente.

4. (hasta **1,5** puntos) Diseño y aplicación de un proceso de **validación** del algoritmo dyv implementado utilizando el método de resolución directa. Hay que indicar claramente los experimentos concretos realizados para asegurarnos de que el programa funciona correctamente, y en su caso programas utilizados para la validación.
5. (hasta **1,5** puntos) **Estudio experimental** del tiempo de ejecución para distintos tamaños de problema. Habrá que experimentar con tamaños suficientemente grandes para obtener resultados significativos ($10^2 \leq n \leq 10^6$). Será necesaria para ello la implementación de un **generador de casos de prueba**, que genere entradas para el programa para los casos más favorable, más desfavorable y promedio.
6. (hasta **1** punto) **Contraste** del estudio teórico y el experimental, buscando justificación a las posibles discrepancias entre los dos estudios.
7. Conclusiones y valoraciones personales de la actividad, con una estimación del tiempo que se ha tardado en completarla (en horas, por cada miembro).

D. Evaluación de la actividad

La documentación generada se enviará mediante una tarea que cada profesor de prácticas abrirá para sus subgrupos. La fecha tope de entrega de la documentación será el 28 de marzo de 2025, durante todo el día.

Los profesores realizarán entrevista con cada uno de los grupos, aunque se puede decidir no realizar la entrevista con alguno de ellos si ha seguido su trabajo en las sesiones de prácticas. La fecha de la entrevista se fijará tras el envío de la documentación.

Respecto a la copia de cualquier ejercicio, de otro grupo o de otra fuente, en particular IAs, será de aplicación el Artículo 22 del Reglamento de Evaluación de Estudiantes (REVA)¹ para el alumnado implicado.

E. Problemas

ALFABETO para todos los problemas: abcdefghijklmnopqrstuvwxyz

NOTA: en los problemas donde aparece $m \leq n$, de cara al análisis (apartados 5-6) puedes asumir $m=100$, pero el programa debe funcionar para cualquier m .

1) Dadas dos cadenas A y B de la misma longitud n , y un natural $m \leq n$, hay que encontrar C, la subcadena de A de tamaño m en la que haya más coincidencias de caracteres entre A y B en sus respectivas posiciones ($A[i]=B[i]$). Devolver el índice p de comienzo de la solución C y el número de caracteres coincidentes. En caso de empate, será válida cualquiera de las soluciones óptimas.

Ejemplo: $n=10$, $m=5$, $A = c d d a b c d a c c$

$B = c a c d d b c a d c$

Solución: C, posición de inicio igual a 6, y número de coincidencias igual a 2.

¹ El o la estudiante que se valga de conductas fraudulentas, incluida la indebida atribución de identidad o autoría, o esté en posesión de medios o instrumentos que faciliten dichas conductas, obtendrá la calificación de cero en el procedimiento de evaluación y, en su caso, podrá ser objeto de sanción, previa apertura de expediente disciplinario.

2) Dadas dos cadenas A y B de la misma longitud n, y un natural $m \leq n$, encontrar C, la subcadena de A de tamaño m con más diferencia total en valor absoluto entre los caracteres en cada posición de A y B (suma de los m valores $|A[i]-B[i]|$ entre las posiciones incluidas en C). Devolver el índice p de comienzo de la solución y el valor de la mayor diferencia total. En caso de empate, será válida cualquiera de las soluciones óptimas.

Ejemplo: n=10, m=5

A= c d d a b c d a c c

B= c a c d d b c a d c

dife: 0 3 1 3 2 1 1 0 1 0

Solución: C, posición de inicio igual a 2, y diferencia total igual a 10.

3) Dada una cadena A de longitud n, y un natural $m \leq n$, encontrar la subcadena B de m caracteres con más caracteres consecutivos ordenados ascendente (incluyendo caracteres iguales). Llamemos C a la subcadena de B formada por dichos caracteres. Devolver como solución el índice p de comienzo B y el número de caracteres en C. En caso de empate, será válida cualquiera de las soluciones óptimas.

Ejemplo: n=10, m=5

A= c d d a b c d a c c

Solución: B, posición de inicio igual a 4, y número de caracteres en C igual a 4 (a b c d).

4) Dada una cadena A de longitud n, y un natural $m \leq n$, hay que encontrar la subcadena B de m caracteres consecutivos con más diferencia total en valor absoluto entre caracteres consecutivos. Devolver el índice p de comienzo de la solución B y dar el valor de la diferencia total máxima. En caso de empate, será válida cualquiera de las soluciones óptimas.

Ejemplo: n=10, m=5

A= c d d a b c d a c c

Diferencias entre caracteres consecutivos: 1 0 3 1 1 1 3 2 0

Solución: B, posición de inicio igual a 5, y diferencia total máxima igual a 7 (1+1+3+2).

5) Dada una cadena A de longitud n, un natural $m \leq n$ y un carácter C, hay que encontrar B, la subcadena de A de tamaño m con más apariciones del carácter C. Devolver el índice p de comienzo de la solución B y el número de veces que aparece C en B. En caso de empate, será válida cualquiera de las soluciones óptimas.

Ejemplo: n=10, m=5, C=c

A= c d d a b c d a c c

Solución: B, posición de inicio igual a 6, y número de apariciones de C = 3.

6) Dada una cadena A de longitud n, un natural $m \leq n$ y un carácter C, hay que encontrar B, la subcadena de A de tamaño m con más apariciones consecutivas del carácter C. Devolver el índice p de comienzo de la solución B y el número de veces que aparece el C consecutivamente en B. En caso de empate, será válida cualquiera de las soluciones óptimas.

Ejemplo: $n=10$, $m=5$, $C=c$

$A = c d d a b c d a c c$

Solución: B, posición de inicio igual a 6, y número de apariciones consecutivas igual a 2.

7) Dada una cadena A con n caracteres y un conjunto S de 5 caracteres distintos, hay que encontrar todas las subcadenas de A formadas por 3 elementos de S sin repetir. Habrá que devolver como solución el número de subcadenas y su posición en la cadena C. Por ejemplo, si

$A = abbfabcddfcbbade$, $n=16$

si consideramos un conjunto de cinco caracteres $S=\{a, b, c, d, e\}$

la solución es 4, en las posiciones 5, 6, 13 y 14

8) Dada una cadena A con n caracteres y un conjunto S de 5 caracteres distintos, devolver todas las subcadenas de A formadas por los 5 elementos de S sin repetir (permutaciones sin repetición). Habrá que obtener el número de subcadenas y su posición en la cadena A. Por ejemplo, si

$A = acbfabcdfcbcbadec$, $n=18$

si consideramos un conjunto de cinco caracteres $S=\{a,b,c,d,e\}$

la solución es 3, en las posiciones 5, 13 y 14

9) Dada una cadena A con n caracteres y 3 subcadenas de longitud 3, devolver las apariciones de concatenaciones de dos de las subcadenas en la cadena A. Obtener el número de subcadenas sus posiciones en la cadena A. Por ejemplo

$A = abbcabcbddacbdcaaac$, $n=18$

y subcadenas son acb, aac, dca,

la solución es 2, en las posiciones 10 (acbdca) y 13 (dcaaac).

10) Sea A una cadena con n caracteres. Sea m un natural tal que $m \leq n$. Sea B cualquier subcadena de A de tamaño m. Y dada una B, sea C la subcadena de B más larga tal que la diferencia en valor absoluto entre sus caracteres consecutivos sea 2 o menos. Se busca la B tal que su C tenga longitud máxima. Indicar el índice p de comienzo de B y el número de caracteres consecutivos de su C. En caso de empate, será válida cualquiera de las soluciones óptimas.

Por ejemplo, si $A = abcehfeksrtzyxdbcehaeksrtzyx$, $m=5$

la posición de inicio de la solución es 1 y el número de caracteres es igual a 4 (abcehfeksrtzyxdbcehaeksrtzyx).