

**Examen de Título**  
**Ciencia de la Computación**

**Fecha:** Martes, 3 de Agosto de 2021

**Tiempo:** 4 horas

**Las Olimpiadas 2020 (?)**

Actualmente se están llevando a cabo las Olimpiadas en Tokio, Japón. Una de las necesidades fundamentales es la de contar con un sistema de información que organice y presente diversos aspectos asociados a dicho evento. Uno de ellos es el medallero. Aparte de ser un elemento que informa al público general respecto del éxito de los deportistas y países, es también un insumo que sirve para analizar resultados por región y país. Futuros cupos de deportistas para el evento por país se determinan parcialmente basado en esos datos.

Por supuesto, junto al medallero deben estar los datos de los deportistas, las disciplinas (por ejemplo, atletismo) que se disputan junto con sus respectivas pruebas (por ejemplo, 100m planos masculino), los resultados de aquellas pruebas, las sedes en que se desarrollan las pruebas y el registro de los récords olímpicos y mundiales.

Para mayor información sobre cómo se organizan disciplinas, pruebas y series (eventos) en las Olimpiadas, puede consultar un par de ejemplos:

- Disciplina: Natación, Pruebas: 400 Combinado Masculino, 100 Mariposa Femenino, 400 Libre Masculino, 400 Combinado Femenino, 100 Braza Masculino, 4x100 Libre Femenino. Se muestran las distintas series de cada prueba además.  
<https://olympics.com/tokyo-2020/olympic-games/es/resultados/natacion/calendario-y-resultados-olimpicos.htm>
- Disciplina: Voley playa, Pruebas: Grupos Femenino y Masculino Preliminares. Se muestran los distintos eventos/series.  
<https://olympics.com/tokyo-2020/olympic-games/es/resultados/voley-playa/calendario-y-resultados-olimpicos.htm>

Respecto del escenario general descrito, se toman los siguientes supuestos:

- Cada prueba específica se asocia a una sola disciplina.
- Asimismo, cada prueba se desarrolla en una única sede de los Juegos Olímpicos actuales.
- Se otorga una medalla de oro, una de plata y una de bronce por prueba y cuentan como una unidad para efectos de su contabilización.
- La contabilización de resultados globales se hace de acuerdo primero a la cantidad de medallas de oro, luego respecto de medallas de plata y finalmente de acuerdo a la cantidad de medallas de bronce.
- Un deportista representa a un único país, pero puede participar en más de una disciplina (por poner un ejemplo, un mismo deportista podría participar en pruebas de basket y handball).
- Cada prueba tiene un récord olímpico y un récord mundial.
- Puede ignorar las distintas fases del desarrollo de cada prueba (eliminatorias, semi-finales y finales) para efectos de los resultados y únicamente quedarse con el resultado al término de desarrollo de la prueba.

Además, como ejemplo, algunos datos interesantes que trabajar y rescatar desde el sistema de información:

- Distribución de atletas por disciplina y por país.
- Las disciplinas practicadas por la mayor cantidad de países.
- Disciplinas que entregan la mayor cantidad de medallas.
- Países que más récords mundiales y olímpicos tienen.

### **P1.-) Modelo de datos (20 ptos Total)**

1. Complete el modelo propuesto con supuestos adicionales, dado que ahora se desea integrar un sistema de admisión de los deportistas a las sedes para competir y un sistema de alojamiento de los deportistas en la villa olímpica. Además complete con supuestos referentes al problema global. (3 ptos)
2. Diseñe un modelo relacional que incluya los elementos generales de almacenamiento considerados para el problema global. (6 ptos)
3. Extienda su modelo relacional e implementelo mediante directivas de SQL para que ahora además considere los elementos del punto 1. de esta pregunta. (3 ptos)
4. Implemente las siguientes consultas en SQL: (2 ptos c/u)
  - a. Nombres y nacionalidades de deportistas que compiten en más de una disciplina, junto con el nombre de esas disciplinas.
  - b. Nombres de países cuyas atletas mujeres obtuvieron más medallas (totales) que los atletas hombres.
  - c. Ranking de países en el medallero.
  - d. Cantidad de países que no tienen un récord olímpico.

### **P2.-) Concurrencia (15 ptos Total)**

Cada una de las pruebas (en que participa un total de  $n$  deportistas), dentro de una disciplina, se desarrolla por series (o eventos) de  $k$  competidores. Esto es, llevan a cabo su prueba grupos de  $k$  competidores al mismo tiempo hasta que los  $n$  deportistas todos hayan completado su prueba. Por ejemplo, en <https://olympics.com/tokyo-2020/olympic-games/es/resultados/atletismo/resultado-100-m-femenino-prel-000100-.htm> se puede ver que hay  $k = 9$  atletas que corren el 100m planos en una serie al mismo tiempo de un total de  $n = 27$  atletas que participan en la prueba. Respecto de este proceder:

1. Implemente un fragmento de código que dirija la organización de una prueba cualquiera en series. (9 ptos)
2. Si cada serie representa un proceso. ¿Qué algoritmo de scheduling utilizaría para calendarizar los procesos? Justifique. (4 ptos) Además indique qué elementos harían que cambiase su respuesta. (2 ptos)

### **P3.-) Estructuras de Datos y Algoritmos (25 ptos Total)**

Diseñe, implemente y analice un algoritmo eficiente (con menor uso de tiempo y espacio posible) para el problema presentado a continuación.

1. Entregue una implementación en el lenguaje de su preferencia entre C, C++, Java y Python. (15 puntos)
2. Justifique que el algoritmo es correcto. (5 puntos)
3. Analice la complejidad en tiempo y espacio del algoritmo. Justifique su eficiencia. (5 puntos)

Las pruebas de natación se realizan en cuatro etapas:

1. clasificatorias - se decide quién participará en los juegos olímpicos,
2. series - se decide quién participará en los semifinales,
3. semifinales - se decide quién participará en las finales,
4. finales - se decide quién ocupará el podio.

Dado que una piscina olímpica tiene 8 pistas, algunas etapas tienen que ser divididas en varios eventos si la cantidad de participantes es superior a 8. Además, dentro de cada evento, hay que asignar una pista a cada uno de los participantes (sea un nadador individual o un equipo de relevos). La asignación de los participantes a diferentes eventos dentro de una misma etapa y a diferentes pistas dentro de un mismo evento se realiza en base a los tiempos alcanzados en la etapa anterior, según uno de los siguientes esquemas.

Se supone que la cantidad de participantes es un múltiplo de 8. Por ejemplo, consideremos una etapa con 24 participantes, ordenados de forma creciente según sus tiempos de la etapa anterior, identificados con las primeras 24 letras del alfabeto inglés ('a' ... 'x'). La cantidad de eventos en una etapa con  $n$  participantes es igual a  $e=n/8$  en general. En este caso,  $e=n/8=3$  - se organizará 3 eventos, con 8 participantes cada uno.

Esquema A

Evento 1							
1	2	3	4	5	6	7	8
g	e	c	a	b	d	f	h
Evento 2							
1	2	3	4	5	6	7	8
o	m	k	i	j	l	n	p
Evento 3							
1	2	3	4	5	6	7	8
w	u	s	q	r	t	v	x

Esquema B

Evento 1							
1	2	3	4	5	6	7	8
s	m	g	a	d	j	p	v
Evento 2							
1	2	3	4	5	6	7	8
t	n	h	b	e	k	q	w
Evento 3							
1	2	3	4	5	6	7	8
u	o	i	c	f	l	r	x

En el Esquema A, en general, al tener  $e$  eventos, los primeros 8 participantes con tiempos más bajos van al Evento 1, los segundos 8 van al Evento 2, etc., hasta el grupo número  $e$  de últimos 8 participantes que van al Evento  $e$ . Dentro de cada evento, los participantes son asignados a las pistas 4, 5, 3, 6, 2, 7, 1, 8 según el orden creciente de su tiempo de la etapa anterior (el participante más rápido es asignado a la pista 4).

En el Esquema B, los primeros  $e$  participantes son asignados a las pistas 4 de todos los Eventos 1- $e$ , siguiendo el mismo orden. Los segundos  $e$  más rápidos son asignados a las pistas 5, etc., hasta los últimos  $e$  participantes que van a las pistas 8.

Se le solicita que desarrolle un programa que tome la información de los participantes de una etapa, con sus tiempos obtenidos en la etapa anterior, y entregue la forma correcta de organizarlos en etapas, primero según el Esquema A, luego según el Esquema B.

El programa debe leer los datos de entrada desde stdin, usando el siguiente formato:

- La primera línea contiene un número entero  $n$ , la cantidad de participantes de la etapa
- Luego vienen  $n$  líneas, una por cada participante. Cada una de estas líneas contiene dos números enteros: el identificador del participante  $id$  y su tiempo obtenido en la etapa anterior  $t$  [en milisegundos] separados por espacio.

El programa debe escribir al stdout la respuesta, usando el siguiente formato:

- La primera línea debe contener **n** números enteros separados por espacios - los identificadores de los participantes ordenados según su asignación a eventos y pistas acorde al Esquema A: primero los participantes asignados a las pistas 1-8 del Evento 1, luego las pistas 1-8 del Evento 2, etc., hasta las pistas 1-8 del Evento  $e=n/8$ .
- La segunda línea debe contener **n** números enteros separados por espacios - los identificadores de los participantes ordenados según su asignación a eventos y pistas acorde al Esquema B (con los detalles del formato iguales al punto anterior).

Restricciones (los datos de entrada siempre los cumplen, no necesitan verificarlas):

- **n** es un múltiplo de 8, entre 8 y 80000, inclusive.
- Para cada participante, su **id** es un entero único entre 1 y **n**, inclusive.
- Para cada participante, su **t** es un entero entre 1 y  $10^7$ , inclusive.

Ejemplo

stdin:

```
16
16 945258
5 974925
6 1025304
1 988751
8 945030
2 999421
14 955708
3 1031388
13 1007447
11 981563
10 969338
15 1029284
4 1055499
12 963144
7 980324
9 941296
```

stdout:

```
5 12 16 9 8 14 10 7 3 6 2 11 1 13 15 4
6 11 12 9 16 5 2 3 15 1 10 8 14 7 13 4
```

### Indicaciones de entrega

- Deberá entregar un único archivo .zip (no se aceptarán otros formatos), una carpeta comprimida con los archivos de respuesta de este examen, con el nombre en el formato que sigue: CS\_<nombre\_apellido>\_n°Archivos.zip
- Las respuestas a las preguntas 1.1, 2.2, 3.2 y 3.3 deberán estar en archivos de formato .txt.
- La respuesta a la pregunta 1.2 será en formato .pdf o .txt.
- Las respuestas a las preguntas 1.3 y 1.4 deberán estar en un formato de script de SQL, .sql. Deberá indicar además el DBMS ocupado en un comentario del script.
- Las respuestas a las preguntas 2.1 y 3.1 deberán estar en formato de texto plano correspondiente al lenguaje de programación seleccionado. Asimismo, deberán ir acompañados de comentarios que indiquen el lenguaje seleccionado e indicaciones de compilación y ejecución.