

Planificador de Conciertos

Javier Larrosa

1 Enunciado

Bruce Springsteen, famoso por sus largos y agotadores conciertos, nos ofrece un verano en su mansión de Malibú a cambio de que le ayudemos a resolver el siguiente problema: En su siguiente gira mundial tiene que decidir qué canciones toca en los conciertos y en qué orden. Para cada canción tenemos su *dificultad* (entero entre 1 y 10), su *ritmo* (lenta, intermedia, rápida), si ha sido o no un "greatest hit" y si exige o no un gran *esfuerzo vocal*. Además, para cada canción tenemos un entero entre 1 y 10 que nos dice cuánto la va a *disfrutar* el público.

Todos los conciertos de la gira van a ser idénticos y constarán de n canciones a escoger de entre las m que forman el repertorio de Bruce. Se tiene que cumplir que: No haya tres canciones consecutivas cuya dificultad sume más de 21; después de una canción que requiera gran esfuerzo vocal tiene que haber como mínimo dos que no lo requieran; No se puede pasar de una canción rápida a una lenta, ni de una lenta a una rápida, directamente; al menos una de cada 4 canciones consecutivas tiene que ser un "greatest hit"; el concierto tiene que empezar y acabar con un "greatest hit".

Lo que nos pide es que le digamos qué canciones y en qué orden tiene que tocar en los conciertos de la gira de manera que se cumplan todas las restricciones anteriores y que hagan disfrutar al máximo al público (la suma de lo que disfruta en cada canción tiene que ser lo mayor posible). No hace falta decir que la misma canción no se puede tocar dos veces en el concierto.

Teneis que respetar la siguiente declaración de datos y variables de decisión:

- enum RITMO = {lenta, intermedia, rapida}
- set of int: REPERTORIO = 1..m;
- set of int: CANCION = 1..n;
- array[REPERTORIO] of boolean: esfVocal;
- array[REPERTORIO] of 1..10: dificultad;
- array[REPERTORIO] of RITMO: ritmo;
- array[REPERTORIO] of boolean: hit;

- `array[REPERTORIO] of 1..10: disfrute;`
- `array[CANCION] of var REPERTORIO: x;`

2 Proyecto

- Diseña un modelo con MiniZinc para este problema.
- Inventa 3 o 4 instancias no triviales para probar tu modelo.

Entrega una carpeta (.zip) que contenga el modelo y las instancias. Ten en cuenta que usare tus instancias tambien para probar los modelos de tus compañeros.

Nota: dos puntos de la practica serán por la calidad de las instancias (quiero instancias que se tarden en resolver entre medio minuto y un par de minutos aproximadamente)