



Universidad Simón Bolívar
CI5437 - Inteligencia Artificial I
Kevin Mena 13-10869
Leonardo López 14-10576
Pedro Samuel Fagundez 15-10460

Proyecto 3

Instrucciones de ejecución:

Para ejecutar el programa se debe correr el comando

```
python3 main.py <archivo_entrada>
```

donde <archivo_entrada> debe ser en formato .JSON.

Al correr el programa se generará un archivo <nombre_torneo>.ics de formato ICalendar.

Solución:

Para la solución primero se lee el JSON de entrada para extraer toda la información del torneo y construir otra que se necesitará para determinar las cláusulas. La información construida es: número de intervalos de juegos (estos son horarios disponibles), número total de intervalos de juegos para todo el torneo, número de juegos que ocurrirán y finalmente el número de variables que tendrá el problema.

Las variables son el número de juegos multiplicado por el total de intervalos de juegos y se dividen de forma tal que se toma en cuenta tres indicadores: el número del participante local, el número del participante visitante y una combinación intervalo-día. Por tanto esta combinación dará un subconjunto específico de variables.

Luego se utiliza la forma en cómo están divididas las variables para calcular las cláusulas, tomando en cuenta las restricciones especificadas.

Las variables están dadas por la siguiente fórmula donde **participante_local** es el índice del participante que juega como local (comenzando por 0), **n_participantes** es la cantidad de participantes, **n_juego** es el índice del juego que el participante juega como local (uno por oponente, yendo de 0 **n_participantes-1**), **n_timeslots** es el número de intervalos de tiempo disponibles y **timeslot** es el intervalo de tiempo para el juego (empezando por 0 que sería el primer intervalo disponible en el primer día y terminando en **n_timeslots-1**).

$$\text{var}(\text{participante_local}, \text{n_juego}) = 1 + \text{participante_local} * (\text{n_participantes} - 1) * \text{n_timeslots} + \text{n_juego} * \text{n_timeslots} + \text{timeslot}$$

Seguido de esto, se traducen a formato CNF para enviar a Glucose e intentar solucionar el problema. Si el problema no tiene solución se aborta el programa, pero si tiene solución se procede a traducir esa solución a eventos del formato ICalendar y colocarlo en un archivo para poder ser cargado en por ejemplo Google Calendar.

Casos de prueba:

Nombre Torneo	Liga Pokémon (Kanto)	Examen Chunin	Examen Chunin	Examen Chunin	Liga Pokémon (Kanto)	Examen Chunin
Duración (Días)	16	23	13	61	15	15
Duración (Horas)	8	8	8	8	8	8
Intervalos por día	4	4	4	4	4	4
Participantes	8	10	10	16	8	10
Número de juegos	56	90	90	240	56	90
Número de variables	3584	8280	4680	58560	3360	5400
Número de cláusulas	596792	1903410	971010	27993840	551992	1145970
¿Satisfacible?	Si	Si	No	Si	Si	No
Conflictos	41152	6195	393253	9944	7304936	16846722
Decisiones	249178	41480	1059717	151622	45459426	50034973
Propagaciones	2629663	328251	61183698	1099500	578681442	2688875327
Tiempo de ejecución	1.06 s	2.23 s	33.69 s	3.49 s	1252.2 s	6747.83 s

Por los resultados obtenidos al correr los casos de pruebas podemos notar que la relación participantes-días es muy importante para la satisfacibilidad o no del problema. Si hay muchos juegos, pero muy pocos días, como es fácil de ver, no se puede resolver el problema porque no hay tiempo físico para que se den todas las restricciones. Otra cosa a notar es el número de variables, si tiende a llegar cerca de los 75.000 ese es el límite en recursos de la representación, es decir, el problema no terminará de correr.