

B003725 Intelligenza Artificiale (2019/20)

Studente: Bernardo Manfredi — <2020-03-31 Tue>

Elaborato assegnato per l'esame finale

Istruzioni generali

Il lavoro svolto sarà oggetto di discussione durante l'esame orale e dovrà essere sottomesso per email due giorni prima dell'esame, includendo:

1. Sorgenti e materiale sviluppato in autonomia (non includere eventuali datasets reperibili online, per i quali basta fornire un link);
2. Un file README che spieghi:
 - come usare il codice per riprodurre i risultati sottomessi
 - se vi sono parti del lavoro riprese da altre fonti (che dovranno essere **opportunamente citate**);
3. Una breve relazione (massimo 4 pagine in formato pdf) che descriva il lavoro ed i risultati sperimentali. Non è necessario ripetere in dettaglio i contenuti del libro di testo o di eventuali articoli, è invece necessario che vengano fornite informazioni sufficienti a *riprodurre* i risultati riportati.

La sottomissione va effettuata preferibilmente come link ad un repository **pubblico** su [github](#), [gitlab](#), o [bitbucket](#). In alternativa è accettabile allegare all'email un singolo file zip; in questo caso è **importante evitare di sottomettere files eseguibili** (inclusi files `.jar` o `.class` generati da Java), al fine di evitare il filtraggio automatico da parte del software antispam di ateneo!

Turni di servizio

In questo esercizio si costruisce un modello per risolvere il seguente problema di soddisfacimento di vincoli. Si devono determinare i turni di servizio di un insieme di n infermieri che lavorano in un ipotetico reparto ospedaliero. Ciascun infermiere deve garantire almeno k ore settimanali per non più di h ore al giorno, comprendendo nelle k ore un turno notturno di 8 ore. Chi effettua il turno notturno ha diritto al riposo per i due giorni successivi. In ogni momento del giorno (6:00-22:00) devono essere presenti in reparto almeno D infermieri ed in ogni momento della notte (22:00-6:00) ne devono essere presenti almeno N .

Il modello deve essere abbastanza generico da permettere di risolvere una qualsiasi istanza. Si sviluppi il modello in un ambiente a scelta tra [MiniZinc](#) e [Numberjack](#) e se ne mostri il funzionamento (con un output intelligibile) su almeno tre istanze distinte.