

B003725 Intelligenza Artificiale (2018/19)

Studente: Umberto Farsetti (5281879) — <2018-12-10 Mon>

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 antispy di ateneo!

Implementazione backtracking e constraint propagation

In questo esercizio si implementa l'algoritmo MAC (maintaining arc consistency) per problemi di soddisfacimento di vincoli descritto in R&N 2009, utilizzando la tecnica AC3 per la propagazione di vincoli. L'algoritmo può essere implementato in un linguaggio di programmazione a scelta. La base del codice dovrebbe essere sufficientemente generale da permettere di risolvere un problema generico ma le descrizioni del dominio e dei vincoli possono essere direttamente codificate nel programma (a differenza di quanto accadrebbe con un risolutore general purpose che sarebbe tipicamente in grado di leggere la specifica del CSP in un linguaggio formale). Il codice sviluppato dovrà essere testato su almeno tre problemi scelti a piacere da [CSPLib](#).