POLITECNICO
DI TORINO

Dipartimento
di Automatica e Informatica

Prerequisiti per Algoritmi e Strutture Dati da Tecniche di Programmazione

Paolo Camurati

- Algoritmo = strategia
- Approccio al problem-solving
- Tipologie di problemi
- Classi di complessità
- Analisi asintotica di complessità di caso peggiore
- Online connectivity
- Matematica discreta
- Ordinamenti

Algoritmo = strategia

- Scegliere un algoritmo spesso significa individuare la miglior strategia per risolvere un problema
- La scelta si basa su:
  - classificazione del problema in base a problemi simili noti
  - conoscenza di algoritmi noti in letteratura
  - conoscenza delle operazioni elementari e delle funzioni di libreria fornite dal linguaggio C
  - conoscenza dei costrutti linguistici (tipi di dato, costrutti condizionali e iterativi)
  - esperienza su tipi diversi di problemi.

Approccio al problem-solving

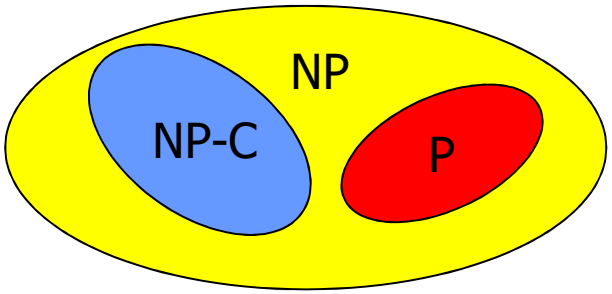
1. analisi del problema: lettura delle specifiche, comprensione del problema, identificazione della classe di problemi noti cui il problema in esame appartiene
2. identificazione delle metodologie: scelta tra paradigmi algoritmici noti (incrementale, divide et impera, programmazione dinamica, greedy, etc.)
3. selezione dell'approccio migliore in base ad un'analisi di complessità
4. decomposizione in sottoproblemi: identificazione dei sottoproblemi e delle loro interazioni in un'ottica di modularità
5. definizione dell'algoritmo risolutivo: identificazione della sequenza di passi elementari e dei dati su cui opera e dimostrazione della correttezza
6. riflessione critica: ripensamento per identificare criticità, migliorie, etc.

Tipologie di problemi

- Problemi di decisione:
  - risposta sì/no, decidibili/non decidibili, trattabili/non trattabili
- Problemi di ricerca:
  - se esiste, identificazione di una soluzione valida
  - ricerca in uno spazio delle soluzioni (con pruning)
  - verifica di validità della soluzione
- Problemi di verifica: verifica di validità della soluzione
- Problemi di ottimizzazione:
  - se esiste, identificazione di una soluzione valida e ottima
  - ricerca in uno spazio delle soluzioni (con pruning)
  - verifica di validità della soluzione
  - enumerazione esaustiva dello spazio delle soluzioni per identificare una soluzione ottima

Classi di complessità

- Classe P: problemi di decisione decidibili e trattabili per cui esiste un algoritmo polinomiale
- Classe NP (Non-deterministico Polinomiale): problemi di decisione decidibili per cui:
  - esistono algoritmi di soluzione esponenziali
  - non conosciamo algoritmi polinomiali, non possiamo però escludere che esistano
  - esistono algoritmi polinomiali di **verifica**
- Classe NP-**C**: problemi di decisione che
  - appartengono a NP
  - tali per cui ogni altro problema in NP è riducibile ad un problema di NP-C attraverso una trasformazione polinomiale


NP
NP-C
P

Analisi di complessità

- Previsione delle risorse (memoria, tempo) richieste dall'algoritmo per la sua esecuzione
- indipendente dalla macchina, asintotica, di caso peggiore
- notazione O, Ω, Θ
- limiti laschi/stretti.

Online Connectivity

Concetti ripresi:

- relazione di equivalenza: componenti fortemente connesse nei grafi orientati
- rappresentazione: algoritmo di Kruskal per gli alberi ricoprenti minimi (Minimum Spanning Trees) in Teoria dei Grafi

Matematica discreta

Sul Portale sono disponibili lucidi che raccolgono ed estendono i concetti relativi a grafi ed alberi introdotti in Tecniche di Programmazione.

Ordinamenti

- Relazione d'ordine
- stabilità/in loco
- limite inferiore alla complessità e ottimalità degli algoritmi
- algoritmo linearitmico: bottom-up mergesort