

Consegna 3

Algoritmo

Lorenzo Dentis, lorenzo.dentis@edu.unito.it

16 marzo 2023

1 Domande

Tenendo conto delle discussioni svolte a lezione sulla definizione di algoritmo e delle sue proprietà scrivete una vostra definizione di algoritmo e delle sue proprietà motivando eventuali differenze rispetto alla formulazione vista a lezione. Nello svolgere questo compito tenete presente che la discussione sulle definizioni deve essere rivolta a classi delle scuole secondarie di secondo grado senza prerequisiti particolari.

2 Risposte

2.1 Definizione di algoritmo

Io ricordo che algoritmo mi fu definito come "una sequenza finita di istruzioni elementari per giungere da un problema ad una soluzione". Ad una ipotetica classe delle scuole secondarie di secondo grado riproporrei questa definizione, perchè riesce ad essere sufficientemente precisa ma è soprattutto molto succinta e facile da memorizzare e comprendere (tanto che ancora adesso la ricordo). Certo non è la definizione più esaustiva e formale, però nella sua semplicità mi piace molto.

Volendo andare nel dettaglio, e basandomi sulla discussione in classe, definirei un algoritmo come *Una sequenza finita e ordinata di istruzioni elementari univoche che da un problema porta alla sua soluzione.*

Non è molto differente dalla definizione data in classe, secondo me è un po' migliore in quanto mette in evidenza il fatto che l'ordine di esecuzione delle istruzioni è molto importante, in secondo luogo credo che questa definizione sia più esplicita riguardo al fatto che l'algoritmo non è la soluzione, l'algoritmo associa un problema alla sua soluzione.

In questo sono della stessa idea del mio collega Cacioli (di cui ho sentito le argomentazioni nelle videoregistrazioni), secondo la mia interpretazione un algoritmo non deve essere corretto per essere considerato tale. Un algoritmo sbagliato è comunque un algoritmo, per due motivi.

1. Per quanto sia possibile verificare formalmente un algoritmo tale operazione non viene quasi mai svolta, quindi potrebbero esserci degli "*edge cases*" che vanno a vanificare la correttezza di un sacco di algoritmi.

mi. Probabilmente tali casi non ci sono, o sono rarissimi, ma senza una verifica formale non si può affermare che un algoritmo sia veramente corretto e basare la definizione di algoritmo su un assunto così forte mi sembra un po' troppo stringente. (Ricordo che il professor Roversi nel corso di Algoritmi e complessità in maniera provocatoria affermava: "Voi avete formalmente dimostrato che il quicksort fornisce una lista ordinata di elementi? No, vi basate sulla fiducia. Siete sicuri che qualcuno lo abbia mai dimostrato?")

2. Volendo essere molto precisi quando si sta parlando di **un** algoritmo ci si riferisce ad un generico algoritmo, non **all'** algoritmo specifico che risolve il problema specifico. Quindi l'algoritmo che non risolve il problema a mio parere è comunque un algoritmo. Non risolve quel problema, magari potrebbe addirittura risolverne un'altro (la professoressa stessa faceva l'esempio dell'algoritmo per la risoluzione di somme a due decimali, che però non funziona a n decimali).

Quindi volendo includere la correttezza tra le proprietà secondo me bisognerebbe specificare che questa è la definizione di *un algoritmo risolutivo per il problema x* .

Naturalmente mi rendo conto che questo secondo punto è veramente un esempio del "voler cercare il pelo nell'uovo" ciononostante credo sia una valida argomentazione.

2.2 Proprietà di un algoritmo

- **Finitezza:** l'algoritmo deve terminare dopo un numero finito di passi
- io suddividerei in 2 parti la proprietà di **Precisione**.
 - Univocità delle istruzioni: Un algoritmo deve essere composto da istruzioni non equivoche.
 - Semplicità delle istruzioni: Le istruzioni devono essere semplici ed eseguibili dall'interprete per cui viene scritto l'algoritmo.
- **Dati in ingresso e in uscita:** Un algoritmo deve accettare zero o più input e deve fornire uno o più output che devono essere dipendenti dagli input immessi.
- **Fattibilità:** L'algoritmo deve essere implementabile, non ci possono essere operazioni astratte e non eseguibili.
- **Efficienza:** I passi dell'algoritmo devono tutti concorrere alla soluzione del problema, non ci devono essere passi che non avanzano la computazione. Nota, ciò è diverso da dire che un algoritmo per essere tale deve essere ottimizzato o efficiente, secondo me anche un algoritmo pessimo è un algoritmo, basta che non faccia operazioni inutili.

Come discusso in precedenza io non inserirei la **correttezza** tra le proprietà di un algoritmo.