

Consegna 4

Problem Solving

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

22 marzo 2023

1 Domande

1. Leggete l'articolo Programming Patterns and Design Patterns in the Introductory Computer Science Course e scrivete un breve resoconto sull'approccio proposto.
2. Riportate il testo del problema analizzato per l'attività **2) Progettazione soluzione/algoritmo** e la soluzione che avete progettato.
3. Descrivete anche eventuali critiche (es. sulla formulazione del problema), problemi e considerazioni emerse durante l'attività.

2 Risposte

2.1 1 Commento sull' articolo

Sono molto d'accordo con il metodo di insegnamento proposto dall'articolo [1], soprattutto perchè è molto affine al metodo che uso io per imparare. Quando incontro una nuova tecnologia o strumento traggo beneficio dal vederlo in uso in un esempio. L'articolo [1] utilizza la stessa procedura in termini di Pattern, presentando un esempio subito dopo aver presentato il problema. Citando il documento (riferimento 4.2):

The Problem Examples section (*n.d.a. seconda sezione*) gives a few examples with only a rudimentary reference to a programming language.

Trovo che i Pattern presentati siano generici il giusto, affermo ciò perchè inconsciamente mi sono trovato ad inserirli nel problema che ci è stato chiesto di sviluppare. Senza sapere che erano Pattern mi sono trovato a pensare: "Questo costruito ritorna spesso in programmazione, potrebbe essere interessante inserirlo nel problema".

Durante la discussione in classe una mia collega aveva sollevato il dubbio che fornire la "soluzione pronta" andasse ad inficiare le capacità di problem solving, in linea generale sono molto d'accordo con la sua affermazione, ma non nel contesto di questo articolo. Infatti, sempre citando dalla sezione 4.2 dell'articolo:

The patterns are written to guide a novice with little or no programming experience who needs a more structured guidance in learning how to program.

Secondo me utilizzare questa metodologia di insegnamento per problemi più sofisticati è controproducente. Una delle capacità più utili di uno sviluppatore è quella del **problem solving** unita all'esperienza di sapere cosa cercare quando si affronta un problema mai visto prima, se istruiamo gli studenti fornendogli Pattern prefabbricati con esempi e soluzioni questi non svilupperanno mai la capacità di trovare da soli una soluzione (anche suboptimale). Insomma l'articolo cerca di risolvere il problema posto dagli studenti *"I do not even know where to start"* fornendo una strategia più guidata alla soluzione, peccato che nella mia esperienza il dubbio "da dove inizio ad affrontare questo problema?" è un dubbio quotidiano quando ho a che fare con l'informatica e si diventa più bravi ad risolvere questo problema solo affrontandolo e fallendo innumerevoli volte. Detto ciò, l'articolo [1] non si propone come panacea per insegnare ogni concetto dell'informatica, bensì si limita ad essere una strategia per *un novizio con poca/nessuna esperienza di programmazione* e secondo me in ciò è eccellente.

2.2 2 Algoritmo di esempio

2.2.1 Algoritmo Dentis

Lo Chef Tony vuole che la sua cucina sia sempre in ordine e soprattutto sapere quali ingredienti ha a disposizione.

Quindi ha suddiviso gli ingredienti in 5 categorie: *pasta, sugo, carne, verdure e spezie*.

Quando arriva un nuovo carico di materie prime (il numero di ingredienti in un carico può variare) Tony vuole che queste vengano separati automaticamente nelle 4 categorie e gli venga restituito in output quanti elementi di ogni categoria ci sono nel carico appena giunto.

Il problema può essere complicato come nella seguente variante Il programma deve tenere conto che Tony non è l'unico Chef di Cat&Ring, nella dispensa potrebbero esserci ingredienti rimasti, quindi deve chiedere allo Chef la quantità rimasta e comportarsi di conseguenza.

2.2.2 Soluzione

2.3 3 Critiche e considerazioni

Riferimenti bibliografici

- [1] *Programming Patterns and Design Patterns in the Introductory Computer Science Course*. URL: <http://www.ccs.northeastern.edu/home/vkp/Papers/Patterns-sigcse2000.pdf>.