

Sommario: 28/2-2/3, 2018

- 1 Quali sono le 4 caratteristiche di un procedimento perchè sia effettivo?
- 2 Si descriva l'algoritmo del quicksort utilizzato in Esercizio L1.1 e si mostri che esso definisce un procedimento effettivo.

2bis Gli algoritmi forniscono procedimenti effettivi ma non eseguibili per le funzioni calcolabili. Questi procedimenti sono utilizzati, come un canovaccio, per la loro implementazione in programmi che a loro volta, forniscono processi automatici e dunque, eseguibili. Si commentino gli altri 2 principali usi, visti nel corso, di tali procedimenti.

- 3 La funzione π_1 , descritta sotto, è calcolabile.

$$\pi_1(n) = \begin{cases} 1 & \text{se l'espansione decimale di } \pi \text{ contiene } n \text{ 1 in sequenza} \\ 0 & \text{altrimenti} \end{cases}$$

- (a) Sapreste fornire una classe di programmi contenente **un** programma per π_1 ?
 - (b) Sapreste giustificare l'affermata calcolabilità di π ?
- 4 Si esamini la definizione data alle pagine 509-510 nell'articolo sulla 3-Colorabilità di N. Jonoska et al., citato a lezione e si risponda:
 - (a) Perché il procedimento in 4 passi, in pag. 510, definisce un algoritmo?

Sommario: 27/2-2/3, 2018

- (b) Sapreste fornire un programma in C che implementi tale algoritmo?
- (c) Sapreste fornire un programma in C che introduca una rappresentazione dei flexible tiles ed emuli le operazioni anneal, ligate, cleave, extract?
- (d) Discutere la relazione tra la soluzione data in (b) e quella data in (c).

5 Sia π , la funzione così definita:

$$\pi(n) = \begin{cases} 1 & \text{se l'espansione decimale di } \pi \text{ contiene } n \\ 0 & \text{altrimenti} \end{cases}$$

- (a) Sapreste fornire una classe di programmi nei quali è certamente contenuto anche **un** programma che calcola π ?
- (b) Sapreste fornire argomenti per affermare, o per negare la calcolabilità di π ?

6 SmallC è un sotto-linguaggio di C senza procedure e costrutti correlati. Si scriva in SmallC un programma che calcoli la funzione *ordinamento di sequenza*.

6bis SmallC è un sotto-linguaggio di C che mantiene tutti i costrutti C fatta eccezione per le procedure e costrutti correlati. Si può scrivere in SmallC un programma che implementa l'algoritmo di quicksort? Si giustifichi la risposta adeguatamente.

Sommario: 27/2-2/3, 2018

- 6ter SmallC è un sotto-linguaggio di C che mantiene tutti i costrutti C fatta eccezione per le procedure e costrutti correlati. Si può dimostrare che $\text{SmallC} \in \text{LP}$. Si dica cosa implica tale fatto?
- 7 L'algoritmo di quicksort definito originariamente da Hoare non pone vincoli sulla: a) dimensione finita della sequenza considerata, nè sul b) tipo di valori formanti la sequenza, nè sulla c) relazione di ordinamento applicata. Di modo che lo stesso algoritmo è applicabile ad ogni sequenza di valori rappresentabili nel linguaggio e su cui sia possibile definire una relazione di ordinamento (anche parziale). Quali di questi vincoli sono presenti nella soluzione data all' Esercizio L1.1 di questo corso?
- 8 (a) Quali tra i seguenti linguaggi sono tra loro collegati: ADA, JAVA, Haskell, SQL, Miranda, Lisp, Smalltalk, ML, C++, Pascal, OCAML?
(b) Per ogni gruppo individuato in (a) si indica natura del collegamento e fonte dell'affermazione.
- 9 Nella diffusione dei linguaggi, richiamata a lezione, sono considerate 4 tipologie di applicazioni.
(a) Si elenchino.
(b) Per ciascuna tipologia si menzioni almeno un'applicazione ed un link web

Sommaro: 27/2-2/3, 2018

- 10 Si consideri il programma QSort2, allegato nei listings della settimana, Listing1, proposto come "soluzione" dell'esercizio L1.1. Si rifrasino le costruzioni utilizzate in modo da completare il programma QSort3 sotto, equivalente a QSort2.

```

int N = ...; //Parametro di programma (costante, non modificabile)
void Scambia(...) {...}
void LeggiSeq(...) {...}
void StampaSeq(...) {...}
void QuickSort(int Left, int Right, int A[]) {...}

int main(void) {
    int Seq[N];
    int k;
    LeggiSeq(Seq, &k); //Lettura sequenza da ordinare
    QuickSort(Seq, k); //Ordinamento
    StampaSeq(Seq, k); //Stampa sequenza ordinata
    return(1);
}

```

- 11 I programmi QSort2 e QSort3 in esercizi L1.1 e L2.10 sono equivalenti e interscambiabili dal punto di vista della funzione calcolata. Ben diverso è invece dal punto di vista dell'espressività. Facendo riferimento a quanto visto a lezione, si dica:

(a) In cosa differiscono QSort2 e QSort3;

Sommario: 27/2-2/3, 2018

- (b) Quale dei due è più espressivo;
 - (c) Quale dei due è più prossimo, e perchè, ad un'implementazione dell'algoritmo di Hoare, richiamato in esercizio L2.7
- 12
- (a) Si discuta una modifica del programma incompleto di esercizio L2.10, sopra, tale che il nuovo programma possa ordinare in accordo ad un ordinamento g su interi, totale ma arbitrario.
 - (b) Si completi il programma ottenuto in (a) in modo da ottenere un programma QSort4 equivalente a QSort2 e QSort3. Si evidenzi definizioni e uso di g .
 - (c) Si compili QSort4 e si esegua il sorgente con sequenza di input 3, -7, 5, -3, 7, opportunamente codificata in accordo alle assunzioni fatte sul formato di I/O del programma
 - (d) Si modifichi QSort4 in un programma che utilizzi un ordinamento: $g(n, m)$ sse $|n| \geq |m|$. In particolare, si mostri la nuova definizione di g , si compili ed esegua il nuovo programma con la sequenza introdotta in (c).
- 13
- (a) Sapreste scrivere un comando, che chiameremo NOP, in C che aggiunto ad un programma in C non modifica la funzione calcolata da tale programma.
 - (b) Quale funzione calcola un programma C costituito dal solo comando NOP?
 - (c) Quanti programmi C calcolano una stessa funzione calcolabile? Si motivi la risposta.