

Appello di Programmazione e Algoritmi 1
15 giugno 2021

Esercizio 1

Cosa stampa il seguente frammento di codice Java?

```
int [] A = {9,2,5,7,3};
int [] B = {6,9,6,2,1};
int conto=a.length;
for (int i=A.length-1;i>=0;i--) {
    if (A[i]<B[i]){
        System.out.println(A[i]+B[i]);
        conto--;
    }
}
System.out.println(conto);
```

NON SARANNO CONSIDERATE VALIDE LE RISPOSTE NON GIUSTIFICATE DAI CALCOLI EFFETTUATI PER OTTENERE IL RISULTATO

Cosa stampa il seguente programma Java?

```
public static void main(String[] args) {
    System.out.println(enigma(2));
    System.out.println(enigma(7));
    System.out.println(enigma(4));
}
public static int enigma (int x){
    if (x<1) return x*x;
    if (x%2==0) return x - enigma(x+1);
    return x+1 + enigma (x-2);
}
```

NON SARANNO CONSIDERATE VALIDE LE RISPOSTE NON GIUSTIFICATE DAI CALCOLI o DAI RAGIONAMENTI EFFETTUATI PER OTTENERE IL RISULTATO

Esercizio 2

Scrivere un metodo iterativo

public static int ProdVett (int[] a, int[] b)

che presi come parametri gli array **a** e **b** di numeri interi restituisce la somma dei prodotti dei termini corrispondenti, cioè

$a[0]*b[0] + a[1]*b[1] + \dots$

Nel caso uno dei due array sia null o vuoto, il risultato sarà 0. Nel caso in cui uno dei due array sia più lungo dell'altro, i termini eccedenti nell'array più lungo non dovranno essere considerati.

Ad esempio, se $a = \{6, 2, 5, 2, 3, 1, 1, 7, 7, 4, 7\}$ e $b = \{3,1,4\}$ allora il risultato è 40 (ovvero $6*3+2*1+5*4$).

Altro esempio: se $a = \{6, 2\}$ e $b = \{3,5,4\}$ allora il risultato è 28 (ovvero $6*3+2*5$).

Si analizzi la complessità temporale del metodo proposto (soluzioni con complessità minore verranno valutate maggiormente).

Esercizio 3

Scrivere un metodo

static int[] copiaFino (int[] a, int k)

che, presi come parametri un array **a** di numeri interi e un numero intero **k** restituisce un nuovo array ottenuto da **a**, considerando tutti gli elementi di **a** dall'inizio fino alla prima occorrenza di **k** escluso. Ad esempio se **a**={10, 9, 2,7,5,1} e **k**=7 dovrà essere restituito l'array {10, 9, 2} (ovvero un array ottenuto copiando gli elementi dell'array **a** dell'inizio fino a quando non si trova il valore di **k**, che in questo caso è 7). Se **a** è null deve essere restituito null. Se **k** non è presente nell'array deve essere restituito un array che ha gli stessi elementi di **a**.

Altri esempi, se **a**={1, 4, 5, 2} e **k**=1 viene restituito l'array vuoto (di lunghezza 0, perché 1 è il primo elemento di **a** e quindi non deve essere copiato nessun elemento di **a**- ricordare che **k**, che in questo caso è 1, non va copiato nell'array risultante).

Se **a**={1, 4, 5, 5} e **k**=2 viene restituito l'array {1, 4, 5, 5} (perché 2 non compare nell'array **a**)

Si analizzi la complessità temporale del metodo proposto.

Esercizio 4

Si consideri il seguente array di numeri interi:

{17,16,3,21,14,22,7,19,13,10}

Mostrare **passo-passo l'esecuzione del merge-sort** per ordinare l'array in modo non decrescente.

Si descriva la complessità computazionale del merge-sort.

[Esercizio di Laboratorio] In relazione al progetto di laboratorio, si implementino i seguenti due metodi:

- int ultimaFigurina(boolean[] album)

restituisce il numero dell'ultima figurina presente nell'album (quella di indice maggiore). Se l'album è vuoto restituisce -1.

- boolean[] creaAlbum(int n, int[] figurine)

restituisce un nuovo album di lunghezza **n** che contenga le figurine i cui indici sono presenti nell'array **figurine**. Ad esempio, con input **n**=7 e l'array {2,3,5} il metodo restituisce l'album:

{false, false, true, true, false, true, false};

Non è necessario implementare il metodo main, ma solamente i due metodi richiesti.

Nell'implementazione dei metodi è possibile sfruttare altri metodi del progetto, se lo ritenete utile.

