

ESERCIZIO ARRAY

Descrizione del problema

Progettare un algoritmo per risolvere il seguente problema. Data una sequenza di interi, verificare se esiste una tripla di elementi consecutivi tali che la loro somma sia un quadrato perfetto. Ad esempio, la sequenza [7, 4, 5, 2] è un'istanza positiva del problema, in quanto $4+7+5=16=4^2$. La sequenza [9, 3, 2, -3] è un'istanza negativa del problema in quanto né la somma degli elementi della prima tripla ($9+3+2=14$), né la somma degli elementi della seconda tripla ($3+2-3=2$) è un quadrato perfetto.

Task

(1 pt) Descrivere in modo sintetico la **specificità del problema**.

(1 pt) Indicare **di che tipo di problema si tratta** (accumulazione, conteggio, verifica esistenziale, verifica universale, ricerca, minimo/massimo).

(3.5 pt) Descrivere un algoritmo risolutivo per il problema utilizzando un **diagramma a blocchi**.

Scrivere un **programma C QuadratiPerfetti** in cui sono definiti:

(6 pt) Una **funzione quadratoPerfetto** con parametri: 1) un array di interi; e 2) un intero che rappresenta la lunghezza dell'array. La funzione restituisce 1 se esiste una tripla della sequenza la cui somma è un quadrato perfetto, 0 altrimenti.

(2.5 pt) Una **funzione main** che gestisce l'interazione con l'utente. La funzione main deve:

- chiedere all'utente quanti interi ha la sequenza e leggere la risposta dell'utente
- chiedere all'utente di introdurre gli interi della sequenza, leggere gli interi introdotti dall'utente e memorizzarli in un array
- invocare la funzione **quadratoPerfetto**, fornendole come parametro un riferimento alla sequenza di interi appena letta e la sua lunghezza
- stampare un messaggio che comunica all'utente se nella sequenza esiste una tripla di elementi consecutivi la cui somma è un quadrato perfetto, oppure no.

Come e cosa consegnare

Consegnare un unico file **main.c** al link "**Consegna Array – C**" su <https://moodle1.ing.uniroma3.it> che, oltre al programma, contiene la specificità ed il tipo di problema (il diagramma a blocchi che illustra l'algoritmo risolutivo deve essere disegnato su un foglio a parte), commentati come segue.

```
/* SPECIFICA
```

```
* Input: ....
```

```
* Pre-condizione: ...
```

```
* Output: ....
```

```
* Post-condizione: ...
```

```
* TIPO DI PROBLEMA: ... */
```

```
#include <stdio.h>
```

```
#include <math.h>
```

```
...
```

ESERCIZIO STRINGHE

Task

Realizzare un'applicazione che consiste di (almeno) tre funzioni.

(6 pt) Una funzione *nienteNumeriDueCifre* riceve come parametro una stringa e la modifica rimuovendo ogni sequenza di esattamente due caratteri numerici consecutivi. Ad esempio, se la stringa ricevuta come parametro è "12ab123fg67xyh2iu09", la stringa deve essere modificata in "ab123fgxyh2iu".

(3 pt) Una funzione *testNienteNumeriDueCifre* verifica la correttezza della funzione *nienteNumeriDueCifre*. La funzione di test deve realizzare un test a scatola nera. Ciascuno degli insiemi di dati di ingresso usato per il test deve essere commentato nel codice con un'indicazione dell'insieme di equivalenza corrispondente. Ad esempio:

```
/* stringa vuota */  
char stringa1[] = "";  
nienteNumeriDueCifre(stringa1);  
printf("Test nienteNumeriDueCifre(\" \"): Atteso = \"\", Calcolato = \"%s\" \n", stringa1);
```

(2 pt) Una funzione *main* chiede all'utente di inserire una stringa da tastiera ed utilizzando la funzione *fgets* memorizza la stringa introdotta dall'utente all'interno di un array di 50 caratteri. Dopo aver rimosso il carattere '\n' dalla stringa, la funzione *main* invoca la funzione *nienteNumeriDueCifre* fornendole come parametro la stringa letta; la funzione *main* stampa quindi la stringa modificata. La funzione *main* termina invocando la funzione di test.

Come e cosa consegnare

Consegnare un unico file **main.c** contenente il programma, al link "**Consegna Stringhe – C**" su <https://moodle1.ing.uniroma3.it>