

Algoritmi e Strutture Dati

Definire il tipo di dato studente, utilizzando il costrutto **struct**, contenente le seguenti componenti:

- matricola: intero
- nome: stringa
- cognome: stringa
- eta: intero

Implementare, inoltre, le funzioni per l'inserimento e la stampa delle componenti della struttura.

Vincoli da rispettare:

- il numero di matricola Ã un numero a sei cifre, che varia fra 255312 e 499999;
- nome e cognome sono stringhe di max 30 caratteri;
- l'etÃ Ã un intero positivo che varia fra 12 e 105.

Suggerimenti

1. Definire un progetto Dev-C++ contenente i file **studente.h**, **studente.cpp** e **provastudente.cpp**
2. Nel file **studente.h** definire la struttura e i prototipi delle funzioni
Stralcio del file **studente.h**:

```
#ifndef _STUDENTE
#define _STUDENTE

struct tipostudente{
    int matricola;
    char nome[30];
    ...
};

typedef struct tipostudente studente;

/* funzioni per l'inserimento dei valori degli elementi della struttura */
int setName(studente *);
int setCognome(studente *);
...
/* funzioni per ottenere i valori degli elementi della struttura */
char * getName(studente *);
...

#endif /* _STUDENTE */
```

3. Nel file **studente.cpp** implementare le funzioni
Stralcio del file **studente.cpp**:

```
#include "studente.h"
...

/* funzioni per l'inserimento dei valori degli elementi della struttura */
...
int setMatricola(studente *Studente){
    ...
```

```
        cin >> Studente->matricola;
        ...
    }

    ...

    /* funzioni per ottenere i valori degli elementi della struttura */
    char * getNome(studente *Studente){
        return (Studente->Nome);
    }
    ...
```

4. Nel file **provastudente.cpp** implementare il main che definisce variabili di tipo **studente** e utilizza le funzioni definite nella libreria **studente.h**.

Risolvere lo stesso esercizio definendo la classe studente (piuttosto che la struttura) avente come metodi le precedenti funzioni.