

**001** – Scrivere un programma C++ che implementi tutti i metodi della classe moneta così definita:

```
class moneta {
private:
    int facce;
    unsigned int seme;
public:
    moneta();
    moneta(unsigned int, int);
    ~moneta();
    void reset(unsigned int);
    int lancio();
    int lanci(int, int*);
};
```

dove:

1. il costruttore di default inizializza gli oggetti della classe moneta assegnando `facce=2` e `seme=0`;
2. il secondo costruttore ha due parametri: il primo è il seme per resettare il PRNG (e viene poi lo assegnato all'attributo `seme` dell'oggetto); il secondo è il numero di facce della moneta e ha (ovviamente) *valore di default* pari a 2 ed è assegnato all'attributo `facce`;
3. il metodo `reset()` prende come parametro il seme per resettare arbitrariamente il PRNG. Questo valore viene poi assegnato all'attributo `seme` della classe;
4. il metodo `lancio()` effettua un singolo lancio della moneta e restituisce 0 (testa) o 1 (croce) in maniera *pseudocasuale*. Se `facce>2`, allora, i possibili risultati vanno da 0 a `facce-1`;
5. il metodo `lanci(int n, int *a)` ha due argomenti: il primo è il numero di lanci della moneta da effettuare, il secondo è il puntatore alla base dell'array in cui sarà riportato l'esito di tutti i lanci (NB l'array è di taglia *almeno* `n` e si assume già dichiarato/allocato dal chiamante)

**002** - Scrivere un programma C++ che implementi tutti i metodi della classe `charnum` così definita:

```
class charnum {
private:
    char *num;
    int len;
public:
    charnum(int);
    ~charnum();
    void show();
    int val();
}
```

dove:

1. Il metodo `charnum(int n)`: alloca un array di caratteri sufficienti a contenere la rappresentazione alfanumerica del numero `n`; assegna al membro `num` il puntatore all'array e a `len` la sua lunghezza; riempie l'array puntato da `num` con i caratteri che compongono la rappresentazione di `n`.

Per esempio, la seguente dichiarazione:

```
charnum centoventisei(126);
```

produce una chiamata al costruttore della classe che: alloca un array di tre caratteri e ne assegna il puntatore a `centoventisei.num`; assegna `centoventisei.len=3`; riempie l'array puntato da `num` con i caratteri `'1'`, `'2'` e `'6'`;

2. Il metodo `show()` visualizza su console il contenuto dell'array;
3. il metodo `val()` restituisce il valore numerico rappresentato dalla lista;

Per esempio, la seguente espressione:

```
centoventisei(126).val()
```

restituisce il valore intero 126;

**Suggerimenti:** Il numero di cifre necessarie a rappresentare un numero  $n$  espresso in base 10 è dato dalla formula  $\text{floor}(\log_{10}(n))+1$ ; Le due funzioni sono fornite dichiarate nell'header file `cmath`. Si rimanda al *manuale in linea di Linux* per la descrizione delle due funzioni (comandi: `man 3 floor` e `man 3 log10`);