BELLA COPIA

CASONATO MATTEO
1227270

②

```
class SIM {
private:
  int tempoTot;
  int traffTot;
  static double costoTelefonate;
  static double costoTraffico;
public:
  virtual double costoMeseCorrente() const = 0
};

double SIM::costoTelefonate = 0,02;
double SIM::costoTraffico = 0,03;

class Tel : public SIM {
private:
  double abbonamentoMensile;
public:
  double costoMeseCorrente() const {
    return abbonamentoMensile + traffTot * costoTraffico;
  }
  double getAbbonamentoMensile() { return abbonamentoMensile; }
};
class Dati : public SIM {
private:
  static double abbonamentoMensile;
public:
  double costoMeseCorrente() const {
    return abbonamentoMensile + tempoTot * costoTelefonate;
  }
};

double Dati::abbonamentoMensile = 19,99;

class PAO {
private:
  std::list<const SIM*> listaSim;
public:
  std::list<Dati*> g1(double x) const {
    std::list<Dati*> listaDati;
    for (const SIM* el : listaSim) {
      if (dynamic_cast<Dati*>(el) && el->costoMeseCorrente() >= x)
      { (*) Sim* el2 = const_cast<Sim*>(el);
        listaDati.add(el2); }
    }
    return listaDati;
  }
```

(*) `Sim* el2 = const_cast<Sim*>(el);`

MACCHIA

CONST

MACC.

pushback

CONST

DOWNCAST

```
std::vector<Tel*> g2() const {
    std::vector<Tel*> v;
    for (const Sim* el : list->Sim) {
        if (dynamic_cast<Tel*>(el) &&
            el->costoMeseCorrente() ≥ el->getAbbonamentoMensile()*2) {
            Sim* el2 = const_cast<Sim*>(el);
            v.push_back(el2);
        }
    }
    return v;
}
}
```

CONST

DOWNCAST

DOWNCAST

4/10

1

| E≤A | E≤B | E≤C | E≤D | D≤A | D≤B | D≤C | C≤A | C≤B |
|---|---|---|---|---|---|---|---|---|
| V | F | V | F | V | F | V | V | F |

| B≤A |
|---|
| V |

7/10

BELLA COPIA

CASONATO MATTEO
1227270

BRUTTA COPIA

CASNATO MATTEO
1227270

①

| E≤A | E≤B | E≤C | E≤D | D≤A | D≤B | D≤C |
|---|---|---|---|---|---|---|
| V | F | V | F | V | F | V |

| C≤A | C≤B | B≤A |
|---|---|---|
| V | F | V |

- C≤A
- !C≤B
- !B≤C
- B≤A
- B≤D
- B≤E
- D≤E

E≤C

②

```
class SIM {
private:
    int tempoTot;
    int traficoTot;
    static double costoTelefonate;
    static double costoTraffico;
public:
    virtual double costoMeseCorrente() = 0;
};

double SIM::costoTelefonate = 0,02;
"      "   costoTraffico  = 0,03;

class Tel : public SIM {
private:
    double abbonamentoMensile;
public:
    double costoMeseCorrente() {
        return abbonamentoMensile + traficoTot * costoTraffico;
    }
};

class Dati : public SIM {
private:
    static double abbonamentoMensile;
public:
    double costoMeseCorrente() {
        return abbonamentoMensile + tempoTot * costoTelefonate;
    }
};

double Dati::abbonamentoMensile = 19,99;
```

# Brutta Copia

```
class PAO {
private:
    std::list<const Sim*> listaSim;
public
```

# PATENTE DI GUIDA REPUBBLICA ITALIANA

1. CASONATO
2. MATTEO
3. 08/08/00 SAN DONA' DI PIAVE (VE)
4a. **19/01/2019** 4c. MC-VE
4b. **08/08/2029**
5. VE5561531K
7.

9. AM B

13.

| 9. | 10. | 11. | 12. |
|---|---|---|---|
| AM | 19/01/19 | 08/08/29 | |
| A1 | | | |
| A2 | | | |
| A | | | |
| B1 | | | |
| B | 19/01/19 | 08/08/29 | |
| C1 | | | |
| C | | | |
| D1 | | | |
| D | | | |
| BE | | | |
| C1E | | | |
| CE | | | |
| D1E | | | |
| DE | | | |
| | | | |
| | | | |

12.

AH 8519006

1. Cognome 2. Nome 3. Data e luogo di nascita 4a. Data del rilascio
4b. Data di scadenza 4c. Rilasciata da 5. Numero della patente
10. Valida dal 11. Valida fino al 12. Codici

ISTITUTO POLIGRAFICO E ZECCA DELLO STATO S.p.A. O.C.V. - ROMA
mod. MC 720P