

## Esercizio Funzione

Si assumano le seguenti specifiche riguardanti la libreria Qt (**ATTENZIONE: non si tratta di codice da definire.**).

- Come noto, `QWidget` è la classe base polimorfa di tutte le classi Gui della libreria Qt. La classe `QWidget` rende disponibile un metodo virtuale `int heightDefault() const` con il seguente comportamento: `w.heightDefault()` ritorna l'altezza di default del widget `w`.
- La classe `QFrame` deriva direttamente e pubblicamente da `QWidget`. La classe `QFrame` rende disponibile un metodo `void setLineWidth(int)` con il seguente comportamento: `f.setLineWidth(z)` imposta la larghezza della cornice del frame `f` al valore `z`.
- La classe `QLabel` deriva direttamente e pubblicamente da `QFrame`.
  - La classe `QLabel` fornisce un overriding del metodo virtuale `QWidget::heightDefault()`.
  - La classe `QLabel` rende disponibile un metodo `void setWordWrap(bool)` con il seguente comportamento: `l.setWordWrap(b)` imposta al valore booleano `b` la proprietà di word-wrapping (andare a capo automaticamente) della label `l`.
- La classe `QSplitter` deriva direttamente e pubblicamente da `QFrame`.
- La classe `QLCDNumber` deriva direttamente e pubblicamente da `QFrame`. La classe `QLCDNumber` rende disponibile un metodo `void setDigitCount(int)` con il seguente comportamento: `lcd.setDigitCount(z)` imposta al valore `z` il numero di cifre dell'intero memorizzato dal `lcdNumber lcd`.

Definire una funzione `fun` di signature `list<QFrame*> fun(vector<QWidget*>&)` con il seguente comportamento: in ogni invocazione `fun(v)`,

1. per ogni puntatore `p` elemento del vector `v`:
  - se `*p` è un `QLabel` allora imposta la larghezza della sua cornice al valore 8 ed imposta a `false` la proprietà di word-wrapping della label `*p`;
  - se `*p` è un `QLCDNumber` allora imposta al valore 3 il numero di cifre dell'intero memorizzato dal `lcdNumber *p`.
2. `fun(v)` deve ritornare una lista contenente **tutti e soli** i puntatori `p` non nulli contenuti nel vector `v` che puntano ad un `QFrame` che non è un `QSplitter` e la cui altezza di default è minore di 10.

```
list<QFrame*> fun(vector<QWidget*>& v){
    list<QFrame*> ret;

    for(auto p = v.begin(); p != v.end(); ++p){
        // sottotipo di QLabel
        QLabel* ql = dynamic_cast<QLabel*>(*p);

        // 1.a
        if(ql){
            ql->setLineWidth(8);
        }
        // alternativa = chiama dentro l'if (lunga)
        if(dynamic_cast<QLabel*>(*p) && dynamic_cast<QLabel*>(*p)->setWordwrap){}

        // 1.b
        QLCDNumber* qlcd = dynamic_cast<QLCDNumber*>(*p);
        if(qlcd){
            ql->setDigitCount(3);
        }

        // 2 (ritorno i QFrame non QSplitter)
        if(dynamic_cast<QFrame*>(*p) && !dynamic_cast<QSplitter*>(*p) &&
            p->heightDefault() < 10){
            v.push_back(p);
        }
    }

    return ret;
}
```