

DV1497: Ominlämningsuppgift C

Generella krav:

Inga variabler får vara globala.

Alla funktioner som kan vara konstanta (const) ska vara det.

Lösningen får inte generera några minnesläckor.

Alla medlemsvariabler ska vara **privata**.

ADT Mängd (Set)

Uppgiften är implementera en klassmall Set.

En mängd är en behållare som håller värden av en viss typ utan inbördes ordning och som inte tillåter dubletter (eller multipler).

En dynamiskt allokerad array av typen T ska användas som intern datastruktur för mängden.

Mängden ska ärva från interfacet ISet nedan – detta får inte modifieras:

```
#ifndef ISET_H
#define ISET_H

template <typename T>
class ISet
{
public:
    virtual bool insert(T element) = 0;
    virtual bool remove(T element) throw(...) = 0;
    virtual int size() const = 0;
};
#endif
```

Beskrivning av medlemsfunktionerna:

- `insert(T element)`: lägger till element till mängden och returnerar true under förutsättning att elementet inte redan finns i mängden (i så fall läggs elementet inte till och false returneras).
Om arrayens kapacitet inte räcker till ska den expandera.
- `remove(T element)`: tar bort element från mängden och returnerar true. Om elementet saknas i mängden returneras false.
Undantag ska hantera situationen ifall mängden är tom.
- `size()`: returnerar antalet element i mängden.

Utöver medlemsfunktionerna ska Du implementera konstruktor, destruktor, kopieringskonstruktor och tilldelningsoperator.

Du ska även bifoga en testfil i vilken Du noggrant testat samtliga publika medlemmar i klassmallen Set.

Klassmallen ska testas för 2 olika datatyper.

Tillse att inga minnesläckor genereras!

Denna fil ska vara väl kommenterad avseende vad som testas var i koden.