

DT019G Objektbaserad programmering i C++

Laboration: Att hantera textposter med hjälp structar.

Martin Kjellqvist, Jimmy Åhlander

structs.tex 209 2017-03-08 13:36:17Z jimahl

Innehåll

1	Introduktion	1
2	Syfte	1
3	Läsanvisningar	2
4	Uppgift	2
5	Examination	3

1 Introduktion

Att hantera data med en naturlig representation är en hörnsten i strukturerad programmering. Dataposter och egendefinierade typer är ett närmast uråldrigt koncept med rötter i COBOL och Pascal.

Att skriva kod som har enkla och naturliga begrepp definierade som typer gör koden lättläst och tillgänglig.

En struct i C++ är numera en lätt förklädd klass. Användningen skiljer sig däremot ganska väsentligt. En struct används som regel för att representera värden. En struct beskriver sällan beteendet hos sina instanser. För en klass är det regelmässigt tvärt om; beteendet har lika stor roll som datat.

2 Syfte

Vi kommer i labben att använda structar för att hantera adresser och fiktiva personuppgifter.

De nya begrepp som används är följande nyckelord `struct`, och konstruktionerna `medlemsvariabel`, `operatoröverlagring`, egen typ som funktionsargument.

Specialiserade funktioner skapas för att behandla datat.

3 Läsanvisningar

Då du studerar material på nätet och i boken måste du ha i åtanke att det inte är någon skillnad på nyckelorden `struct` och `class` annat än att medlemmarna i en `struct` är synliga om inget annat angivits. Bortse från nyckelorden `private` och `public` i denna lab.

Du bör ha läst om följande datastrukturer i standardbiblioteket för C++:

- `std::string`,
- `std::vector`.

Därefter bör du ha läst om följande funktioner från standardbibliotekets algoritm-del:

- `std::find`,
- `std::count`,

Även om följande funktioner från C-biblioteket:

- `tolower(3)`,
- `toupper(3)`.

Du kan läsa om standardbiblioteket i kurslitteraturen [2] eller någon referens, exempelvis *C++ Reference* [1] eller manualsidorna.

Att överlagra inströmsoperatoren finner du i kurslitteraturen under ”overloaded operator, input”.

4 Uppgift

Formulera en konsollapplikation som läser in uppgifterna i textfilen `names.txt` till en struktur `person` som har följande medlemsvariabler:

```
name:std::string, id:std::string, location:address.
```

En person har som medlem `location` en egendefinierad typ `address` med följande medlemmar.

```
street:std::string, zip:int, city:std::string
```

Inläsningen ska ske till en `vector<person>`.

Det krävs att ditt program innehåller följande funktioner.

```
/**
 * Read from file <filename> with the format
 * Name\n
 * ID\n
 * Street data, 5 digit zip[2 SPACE]city\n
 * @returns vector<person>. If file is not found, it returns
 *   an empty vector.
 */
std::vector<person> read_file(string filename);
/**
```

```

* Look in vector for names containing the substring
  name_part.
* The search is case insensitive.
* @returns the number of occurrences name_part is found.
*/
size_t find_in_names(vector<person> persons, string
  name_part);

/**
* Look in vector for persons living in a particular city.
* Exact matches only.
* The search is case insensitive.
* @returns a vector containing the matched persons.
*
vector<person> find_person_from_city(vector<person>
  haystack, string city);

```

Huvudprogrammet består av en meny med 3 alternativ

```

1 - Sök del av personnamn.
2 - Sök städer.
3 - Avsluta.
:

```

Användaren anger ett av alternativen och får därefter mata in söksträngen. Alternativ 1 ska endast skriva ut antalet funna namn på en egen linje, därefter visa menyn igen. [heltal+nyrad].

Alternativ 2 ska endast skriva ut personens namn och ortnamnet på en rad, därefter visa menyn. [namn, ortnamn+nyrad].

Inläsningen ifrån textfilen ska göras genom att överlagra inströmsoperatören.

```

istream& operator>>(istream& in, person& p);

```

5 Examination

Din lösning ska redovisas muntligen för en lärare vid något av kursens redovisningstillfällen. När du redovisat och fått godkänt laddar du upp din källkod (och byggsript) till inlämningslådan i lärplattformen.

Referenser

- [1] *C++ Reference*, 2017. URL <http://www.cppreference.com/>.
- [2] Lippman, Stanley, Moo, Barbara, och Lajoie, Josée. *C++ Primer*. Addison Westley, New York, 5:e internationella utgåvan, 2012.