

Inlämningsuppgift A 2024-12-10

Du ska implementera totalt 4 funktioner som alla måste vara korrekt lösta. Du tillför funktions-deklarationerna i `Todo.h` och motsvarande funktions-definitioner i `ToDo.cpp`.

I filen `TestCorrectness.cpp` finns tester som används för att kontrollera din lösning, dvs testar de funktioner du implementerat.

Det är inte tillåtet att inkludera andra bibliotek än de som redan är inkluderade.

Funktion som dubblar värdet på ett heltal

Funktionens namn ska vara **`doubleUp`**.

Parameter ska anges så att det överensstämmer med anropen som visas i exemplet nedan där **`number`** är heltalsvariabel.

Ex 1:

`number` har värdet 200 innan anropet

`doubleUp(number);`

efter anropet har **`number`** värdet 400.

Ex 2:

`number` har värdet -22 innan anropet

`doubleUp (number);`

efter anropet har **`number`** värdet -44.

Funktion som läser från textfil

Läs ett antal strängar från en textfil. Dessa ska placeras i en array. Antalet som ska läsas ges som parameter.

Du kan förutsätta att det finns tillräckligt med strängar på filen och att arrayen är tillräckligt stor. På varje rad i filen finns en sträng. Strängarna kan innehålla blank-tecken (ex-vis mellanslag).

Funktionens namn ska vara **`readStringsFromFile`**.

Parametrar ska anges så att det överensstämmer med anropet som visas i exemplet nedan.

"example.txt" är namnet på textfilen med strängar och **`strArr`** är arrayen och 5 är det antal som ska läsas och placeras i arrayen.

`readStringsFromFile("example.txt", strArr, 5);`

Efter anropet innehåller **`strArr`** 5 strängar, placerade från och med index 0 och framåt, som har lästs från filen med namnet "example.txt".

Funktion som bestämmer hur många heltal i en array som är större än ett visst heltalsvärde

Antalet ska returneras.

Om inget specifikt heltalsvärde anges ska det vara 0.

Funktionens namn ska vara **nrOfValuesBiggerThan**.

Parametrar ska anges så att de överensstämmer med anropen som visas i exemplet nedan, där **arr** representerar en array med heltal och **nrOf** representerar antalet heltal i arrayen.

Ex 1:

```
int result = nrOfValuesBiggerThan (arr, nrOf, 50);
```

result innehåller efter anropet det antal heltal i arrayen **arr** som är större än 50.

Ex 2:

```
int result = nrOfValuesBiggerThan (arr, nrOf);
```

result innehåller efter anropet det antal heltal i arrayen **arr** som är större än 0.

Funktion som bestämmer index för det minsta och index för det största heltalet i en array med heltal

Funktionens namn ska vara **lowestAndHighest**.

Parametrar ska anges så att de överensstämmer med anropen som visas i exemplet nedan, där **arr** representerar en array med heltal, **nrOf** representerar antalet i arrayen, **indexOfLowest** och **indexOfHighest** är heltalsvariabler.

Ex 1:

arr har innehållet 23, 55, -11, 5, 77, -8 och **nrOf** har värdet 6

```
lowestAndHighest(arr, nrOf, indexOfLowest, indexOfHighest);
```

efter anropet innehåller **indexOfLowest** värdet 2 och **indexOfHighest** innehåller värdet 4

Ex 2:

arr har innehållet 22, 22, 22 och **nrOf** har värdet 3

```
lowestAndHighest(arr, nrOf, indexOfLowest, indexOfHighest);
```

efter anropet innehåller **indexOfLowest** värdet 0 och **indexOfHighest** innehåller värdet 0