

(https://profile.intra.42.fr)

# FT\_CONTAINERS (/PROJECTS/FT\_CONTAINERS) PROJESİ İÇİN ÖLÇEKLENDİRME (/PROJECTS/FT\_CONTAINERS)

Bu takımdaki 1 öğrenciyi değerlendirmelisiniz.



Git deposu

git@vogosphere.42kocaeli.com.tr:vogosphere/intra-uuid-1641620



## giriş

Lütfen aşağıdaki kurallara uyun:

- Değerlendirme süreci boyunca kibar, nazik, saygılı ve yapıcı kalın  
. Toplumun refahı buna bağlıdır.

- Çalışmaları değerlendirilen öğrenci veya grupla  
projelerindeki olası işlev bozukluklarını tanımlayın.

Tanımlanmış olabilecek sorunları tartışmak ve tartışmak için zaman ayırın . - Meslektaşlarınızın projenin yönergelerini ve işlevlerinin kapsamını

nasıl anlamış olabileceği konusunda bazı farklılıklar olabileceğini göz önünde bulundurmalısınız . Her zaman açık fikirli olun ve bunları mümkün olduğunca dürüst bir şekilde derecelendirin. Pedagoji yalnızca ve yalnızca ekran değerlendirmesi ciddi bir şekilde yapıldığında yararlıdır.

## Yönergeler

- Yalnızca değerlendirilen öğrenci veya grubun Git deposunda döndürülen çalışmaya not verin

- Git deposunun öğrenciye/öğrencilere ait olduğunu tekrar kontrol edin.

Projenin beklenen proje olduğundan emin olun . Ayrıca, boş bir klasörde 'git clone' kullanılıp kullanılmadığını kontrol edin

. - Sizi kandırmak ve resmi deponun içeriği olmayan bir şeyi değerlendirmenize

neden olmak için kötü niyetli takma adların kullanılmadığından emin olun . - Sürprizlerden kaçınmak için ve mümkünse, derecelendirmeyi kolaylaştırmak için kullanılan komut dosyalarını (test veya otomasyon komut dosyaları) birlikte inceleyin. - Değerlendireceğiniz ödevi tamamlamadıysanız, değerlendirme sürecine başlamadan önce konunun tamamını okumanız gerekmektedir.

- Boş bir depoyu, çalışmayan bir

programı, bir Norm hatasını, hile yapmayı vb. bildirmek için mevcut bayrakları kullanın.

Bu durumlarda değerlendirme süreci sona erer ve final notu 0,

kopya durumunda -42 olur. Ancak, kopya çekme dışında, öğrencilerin gelecekte tekrarlanmaması gereken hataları belirlemek için

teslim edilen çalışmaları birlikte gözden geçirmeleri şiddetle tavsiye edilir . - Varsa, yapılandırma dosyası dışında herhangi bir dosyayı asla düzenlemek zorunda kalmamalısınız . Bir dosyayı düzenlemek istiyorsanız, değerlendirilen öğrenciyle nedenlerini açıklamak için zaman ayırın ve ikinizin de bu konuda uygun olduğundan emin olun. - Ayrıca bellek sızıntısı olmadığını da doğrulamanız gerekir. Ayrılan herhangi bir bellek

yığın, yürütmenin bitiminden önce uygun şekilde serbest bırakılmalıdır. Sızıntılar, valgrind veya e\_fence gibi bilgisayarda bulunan farklı araçlardan herhangi birini kullanmanıza izin verilir .

Bellek sızıntısı durumunda

uygun bayrağı işaretleyin.

---

## Ekler



konu.pdf (<https://cdn.intra.42.fr/pdf/pdf/60315/en.subject.pdf>)



ana.cpp (/uploads/document/document/10932/main.cpp)

## Ön testler

*Hile yapıldığından şüpheleniliyorsa, değerlendirme burada durur. Bildirmek için "Hile" bayrağını kullanın. Bu kararı sakince, akıllıca alın ve lütfen bu düğmeyi dikkatli kullanın.*

## Önkoşullar

Kod, c++ ve -Wall -Wextra -Werror bayraklarıyla derlenmelidir.

Bir Makefile var, proje gerekli seçeneklerle doğru bir şekilde derleniyor, C++ ile yazılmış.

Kod ayrıca -std=c++98 bayrağıyla da derlenmelidir.

Don't forget this project has to follow the C++98 standard. Thus, C++11 (and later) functions or containers are NOT expected.

Any of these means you must not grade the exercise in question:

- A function is implemented in a header file (except for template functions).
- A Makefile compiles without the required flags and/or another compiler than c++.

Any of these means that you must flag the project with "Forbidden Function":

- Use of a "C" function (\*alloc, \*printf, free).
- Use of a function not allowed in the exercise guidelines.
- Use of "using namespace <ns\_name>" or the "friend" keyword (in this project, the "friend" keyword is allowed for specific uses, you'll see).
- Use of an external library, or features from versions other than C++98.
- Use of a STL container to implement another container.

✓ Yes

✗ No

---

## Mandatory part

Verify that each mandatory container is correctly implemented. If the vector or the map is missing, check the 'Incomplete work' flag. But note the project can be passed without the stack. Do not hesitate to ask to the evaluated student what is the reference they used. A main.cpp is available to download on this page. It should compile with the student's containers. The evaluated student has also to turn in their own tests.

---

### Vector - Basics

Make sure that every member function, non-member function and overload is present and works as expected. Same thing for iterators.

Use the STL container to check that everything works the same way.

✓ Yes

✗ No

---

### Vector - Advance

- The inner data structure should be a dynamic array.
- const\_iterator and iterators should be comparable.
- Check the dynamic reallocation system.

- Test the `swap()` function: After swap, all the iterators, pointers and references pointing to elements in both containers remain valid and are now pointing to the same elements than before the call but in the other container (where they now iterate).
- Check that the friend keyword is used only for the relational operators.

✓ Yes

✗ No

### Vector - Performance

Make sure that the speed is reasonable compared to the STL container!

For example, a deep copy should allocate all the memory in one call.

✓ Yes

✗ No

### Map - Basics

Make sure that every member function, non-member function and overload is present and works as expected. Same thing for iterators.

Use the STL container to check that everything works the same way.

✓ Yes

✗ No

### Map - Advance

- Check the inner structure. It should be an ordered tree (AVL tree, Red-Black tree, and so forth). For example, a simple array is not ok.
- Verify that `pair<>` is recoded and used.
- Verify that `ft::make_pair()` behaves as expected.
- There mustn't be several identical keys. Each key must be unique in map.
- Check that the keys are sorted using the comparison function (see Compare).
- Check `std::allocator` and `allocator::rebind` are used and there's no direct usage of `new` (rebind could be used in the tree).
- Check that insertion and deletion (erase) do not invalidate iterators.
- The `swap()` function should not move data but only pointers.
- Check that the friend keyword is used only for the relational operators and `map::value_compare`.
- There's no memory leak.

✓ Yes

✗ No

### Map - Performance

Make sure that the speed is reasonable compared to the STL container!

Slower than the STL map is ok.

A complete timeout is not.

If it's more than 20 times slower than the map STL, count it false.

☒ Yes☐ No

### Stack - Basics

Make sure that every member function, non-member function and overload is present and works as expected.

Use the STL container to check that everything works the same way.

☒ Yes☐ No

### Stack - Advance

- The default underlying container must be the vector of the student.
- The underlying container must be protected and not private!
- The standard containers vector, deque and list are compatible as underlying containers.
- The stack cannot be iterate.

☒ Yes☐ No

## Bonus part

*Evaluate the bonus part if, and only if, the mandatory part has been entirely and perfectly done, and the error management handles unexpected or bad usage. In case all the mandatory points were not passed during the defense, bonus points must be totally ignored.*

### Set

Make sure that every member function, non-member function and overload is present and works as expected. Same thing for iterators.

Use the STL container to check that everything works the same way.

The inner data structure must be a Red-Black tree.

Ask the evaluated student for details. How does it work? If you have any doubt, don't validate this bonus as the set is easier than the map.



The only bonus here is for the Red-Black tree!

☒ Yes☐ No

## Ratings

Don't forget to check the flag corresponding to the defense

☒ Ok☒ seçkin proje☐ boş iş☐ tamamlanmamış iş☐ geçersiz derleme☐ Aldatmak☐ Kaza☐ ilgili durum

 Sızıntılar yasak işlev

## Çözüm

Bu değerlendirme için bir yorum bırakın

Değerlendirmeyi bitir

Video gözetimi için kullanım koşulları (<https://profile.intra.42.fr/legal/terms/1>)

Çerezlerin kullanımına ilişkin beyan (<https://profile.intra.42.fr/legal/terms/2>)

Sitenin genel kullanım şartları (<https://profile.intra.42.fr/legal/terms/6>)

Yasal uyarılar (<https://profile.intra.42.fr/legal/terms/3>)

İç Yönerge (<https://profile.intra.42.fr/legal/terms/8>)

Gizlilik Politikası (<https://profile.intra.42.fr/legal/terms/5>)