

CPP MODÜL 02 PROJESİ İÇİN ÖLÇEK

(/PROJECTS/CPP-MODULE-02)

Bu takımda 1 öğrenciyi değerlendirmelisiniz

Git repository

Giriş

Lütfen aşağıdaki kurallara uyun:

- Değerlendirme süreci boyunca kibar, nazik, saygılı ve yapıçı kalın. Topluluğun refahı buna bağlıdır.
- Çalışması değerlendirilen öğrenci veya grupla projelerindeki olası aksaklıkları belirleyin. Tespit edilmiş olabilecek sorunları tartışmak ve müzakere etmek için zaman ayırın.
- Akranlarınızın proje talimatlarını ve işlevselliklerinin kapsamını nasıl anlamış olabileceği konusunda bazı farklılıklar olabileceğini göz önünde bulundurmalısınız. Her zaman açık fikirli olun ve onları mümkün olduğunda dürüstçe değerlendirin. Pedagoji yalnızca ve yalnızca akran değerlendirmesi ciddi şekilde yapıldığında yararlıdır.

Kılavuz İlkeler

- Yalnızca değerlendirilen öğrenci veya grubun Git repository'sinde teslim edilen çalışmayı değerlendirin.
- Git repository'sinin öğrenci(ler)e ait olduğunu iki kez kontrol edin. Projenin beklenen proje olduğundan emin olun. Ayrıca, 'git clone'un boş bir klasörde kullanıldığını kontrol edin.
- Sizi kandırmak ve resmi repository'nin içeriği olmayan bir şeyi değerlendirmenizi sağlamak için herhangi bir kötü niyetli alias kullanılmadığını dikkatli bir şekilde kontrol edin.
- Herhangi bir sürprizden kaçınmak için ve uygunsa, değerlendirme kolaylaştmak için kullanılan script'leri birlikte gözden geçirin (test veya otomasyon için script'ler).
- Değerlendireceğiniz ödevi tamamlamadıysanız, değerlendirme sürecini başlatmadan önce tüm konuyu okumalısınız.
- Boş repository, çalışmayan program, Norm hatası, kopya ve benzeri durumları bildirmek için mevcut flag'leri kullanın.

Bu durumlarda, değerlendirme süreci sona erer ve final notu 0 olur, kopya durumunda -42 olur. Ancak, kopya dışında, öğrenciler gelecekte tekrarlanmaması gereken hataları belirlemek amacıyla teslim edilen çalışmayı birlikte gözden geçirmeye güçlü bir şekilde teşvik edilir.

- Varsa yapılandırma dosyası dışında hiçbir dosyayı düzenlememeniz gereklidir. Bir dosyayı düzenlemek istiyorsanız, nedenlerini değerlendirilen öğrenciye açıklayın ve ikinizin de bu konuda hemfikir olduğundan emin olun.
- Ayrıca memory leak'lerin yokluğunu da doğrulamalısınız. Heap'te ayrılan herhangi bir memory, yürütme sonundan önce düzgün şekilde serbest bırakılmalıdır.

Leaks, valgrind veya e_fence gibi bilgisayarda mevcut farklı araçlardan herhangi birini kullanabilirsiniz. Memory leak durumunda, uygun flag'i işaretleyin.

Ekler



Ön Testler

Kopya şüphesi varsa, değerlendirme burada durur. Bunu bildirmek için "Cheat" flag'ını kullanın. Bu kararı sakin, akıllıca alın ve lütfen bu düğmeyi dikkatli kullanın.

Ön Koşullar

Kod c++ ve -Wall -Wextra -Werror flag'leri ile derlenmelidir.

Bu projenin C++98 standardını takip etmesi gerektiğini unutmayın. Bu nedenle, C++11 (ve sonrası) function'ları veya container'ları BEKLENMEMEKTEDİR.

Bunlardan herhangi biri, söz konusu alıştırmayı değerlendirmemeden gerektiği anlamına gelir:

- Bir function header dosyasında implement edilmiş (template function'lar hariç).
- Makefile gerekli flag'ler olmadan ve/veya c++ dışında başka bir compiler ile derleniyor.

Bunlardan herhangi biri, projeyi "Forbidden Function" ile işaretlemeniz gerektiği anlamına gelir:

- "C" function'ının kullanımı (*alloc, *printf, free).
- Alıştırma kılavuzlarında izin verilmeyen bir function'ın kullanımı.
- "using namespace" veya "friend" keyword'ünün kullanımı.
- External library veya C++98 dışındaki versiyonlardan özellik kullanımı.

Ex00: Orthodox Canonical Formda İlk Class'ım

Bu alıştırma basit bir aritmetik örnekle canonical class kavramını tanıtır: fixed-point sayılar.

Makefile

Evet Hayır

Uygun flag'lerle derleyen bir Makefile var mı?

Accessor'lar

Evet Hayır

Fixed class (veya adı her neyse) ham değere accessor'lar sağlamalıdır:

- int getRawBits(void) const;
- void setRawBits(int const raw);

Bu member function'lar mevcut ve işlevsel mi?

Canonical

Evet Hayır

Canonical bir class en az şunları sağlamalıdır:

- Default constructor
- Destructor
- Copy constructor
- Copy assignment operator

Bu öğeler mevcut ve işlevsel mi?

Ex01: Daha kullanışlı bir fixed-point number class'ına doğru

Ex00 iyi bir başlangıçtı, ancak class'ımız sadece 0.0 fixed-point değerini temsil edebildiği için hâlâ oldukça işe yaramaz.

Makefile

Evet Hayır

Uygun flag'lerle derleyen bir Makefile var mı?

Floating-point constructor

Evet Hayır

Floating-point değerden bir instance oluşturmak mümkün mü?

<< operator

Evet Hayır

<< operator overload'i var mı ve işlevsel mi?

Fixed-point değeri integer değere

Evet Hayır

Fixed-point değeri integer değere dönüştüren "int tolnt(void) const;" member function'ı mevcut olmalıdır. İşlevsel mi?

Fixed-point değeri floating point değere

Evet Hayır

Fixed-point değeri float değere dönüştüren "float toFloat(void) const;" member function'ı mevcut olmalıdır. İşlevsel mi?

Integer constructor

Evet Hayır

Integer değerden bir instance oluşturmak mümkün mü?

Ex02: Şimdi konuşuyoruz

Bu alıştırma class'a karşılaştırma ve aritmetik özellikler ekler.

Makefile

Evet Hayır

Uygun flag'lerle derleyen bir Makefile var mı?

Comparison operator'ları

Evet Hayır

Altı comparison operator (>, <, >=, <=, == ve !=) implement edildi ve düzgün çalışıyor mu?

Arithmetic operator'ları

Evet Hayır

Dört arithmetic operator (+, -, * ve /) implement edildi ve düzgün çalışıyor mu?

Diğer operator'lar

Evet Hayır

Pre-increment, post-increment, pre-decrement ve post-decrement operator'ları implement edildi ve düzgün çalışıyor mu?

Static member function'ların overload'ları

Evet Hayır

Son olarak, min() ve max() static member function'larının implement edildiğini ve düzgün çalıştığını test edin.

Ex03: BSP

Bu alıştırma, temel unsurlar amaçlandığı gibi çalışlığında karmaşık algoritmaları implement etmenin ne kadar kolay olduğunu fark ettirmiş olmalıdır.

Makefile

Evet Hayır

Uygun flag'lerle derleyen bir Makefile var mı?

Point Class'i

Evet Hayır

Fixed const tipinde iki attribute'a (x ve y) sahip Point class'i var mı?

Ayrıca iki float alan ve x ile y'yi bu değerlerle initialize eden bir constructor'i var mı?

bsp Function'i

Evet Hayır

"bool bsp(Point const a, Point const b, Point const c, Point const point)" prototype'ına sahip bsp() function'i var mı?

Function, nokta a, b ve c köşeleri tarafından tanımlanan üçgenin içindeyse True döndürür.

Aksi takdirde False döndürür.

Main ve testler

Evet Hayır

bsp() function'ının gerektiği gibi çalıştığını test eden en az bir main var mı?

Döndürülen değerin doğru olduğundan emin olmak için birkaç test çalıştırın.

Değerlendirmeyi Bitir

Savunmaya karşılık gelen flag'i kontrol etmeyi unutmayın:

Tamam

Boş çalışma Eksik çalışma Geçersiz derleme Kopya Çöküş

Yasak function

Sonuç

Bu değerlendirme hakkında yorum bırakın

Alt bilgiler:

- Gizlilik politikası
- Video gözetim kullanım koşulları
- İç tüzük
- Cookie kullanımı beyanı
- Sitenin genel kullanım koşulları
- Yasal uyarılar