

[Q \(https://profile.intra.42.fr/searches\)](https://profile.intra.42.fr/searches)

gsever

<https://profile.intra.42.fr>

PROJE CPP MODÜLÜ 02 İİN ÖLEK (/PROJECTS/CPP-MODULE-02)

Bu takımdaki 1 öđrenciyi deđerlendirmelisiniz



Git deposu

`git@vogsphere.42kocaeli.com.tr:vogsphere/intra-uuid-02205`



giriş

Lütfen aşıđıdaki kurallara uyunuz: - Deđerlendirme süreci

boyunca kibar, nazik, saygılı ve yapıcı olun.
Toplumun refahı buna bađlıdır.

- alıřmaları deđerlendirilen öđrenci veya grupla
projelerindeki olası işlev bozukluklarını belirleyin.
Belirlenmiş olabilecek sorunları tartışmak ve tartışmak için zaman ayırın . - Meslektaşlarınızın projenin
talimatlarını ve işlevlerinin kapsamını

nasıl anlamış olabileceđi konusunda bazı farklılıklar olabileceđini göz önünde bulundurmalısınız . Her zaman
açık fikirli olun ve onları mümkün olduđuunca dürüst bir şekilde deđerlendirin. Pedagoji, ancak ve ancak akran
deđerlendirmesi ciddi bir şekilde yapıldığında faydalıdır.

Yönergeler

- Yalnızca deđerlendirilen öđrenci veya grubun Git deposunda döndürülen alıřmaya not verin .

- Git deposunun öđrenciye/öđrencilere ait olduđunu iki kez kontrol edin.

Projenin beklenen olduđundan emin olun .

Ayrıca, boş bir klasörde 'git klonunun' kullanıldığını kontrol edin . - Sizi kandırmak ve resmi arşivin içeriđi
olmayan bir şeyi deđerlendirmenizi

sađlamak için hiçbir kötü niyetli takma adın kullanılmadıđını dikkatlice kontrol edin . - Sürprizlerden kaçınmak
iin ve varsa , not vermeyi kolaylařtırmak için kullanılan komut dosyalarını (test veya otomasyon komut
dosyaları) birlikte inceleyin. - Deđerlendireceđiniz ödevi tamamlamadıysanız, deđerlendirme sürecine

başlamadan önce konunun tamamını okumanız gerekmektedir.


- Boş bir depoyu, çalışmayan bir programı, bir Norm hatasını, hileyi vb. bildirmek için mevcut bayrakları kullanın .

Bu durumlarda değerlendirme süreci sona erer ve final notu 0, kopya durumunda ise -42 olur. Ancak kopya dışında, öğrencinin teslim ettiği çalışmayı birlikte gözden geçirmesi ve gelecekte tekrarlanmaması gereken hataları tespit etmesi şiddetle tavsiye edilir.

- Varsa, yapılandırma dosyası dışında herhangi bir dosyayı asla düzenlemeniz gerekmemelidir. Bir dosyayı düzenlemek istiyorsanız, nedenleri değerlendirilen öğrenciye açıklamak için zaman ayırın ve ikinizin de bu konuda uygun olduğundan emin olun.

- Ayrıca bellek sızıntısı olmadığını da doğrulamalısınız. 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 olması durumunda uygun bayrağı işaretleyin.

ekler

 konu.pdf (<https://cdn.intra.42.fr/pdf/pdf/52150/en.subject.pdf>)

Ön testler

Hile şüphesi varsa değerlendirme burada durur. Bunu bildirmek için "Hile" bayrağını kullanın. Bu kararı sakince, akıllıca alın ve lütfen bu düğmeyi dikkatli kullanın.

Önkoşullar

The code must compile with c++ and the flags -Wall -Wextra -Werror
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 " or the "friend" keyword.
- Use of an external library, or features from versions other than C++98.

☒ Yes☐ No

Exercise 00: My First Class in Orthodox Canonical Form

This exercise introduces the notion of canonical class with a simple arithmetic example: the fixed-point numbers.

Makefile

There is a Makefile that compiles using the appropriate flags.

☒ Yes☐ No

Accessors

The Fixed class (or whatever its name) must provide accessors to the raw value:

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

Are these member functions present and functional?

☒ Yes☐ No

Canonical

A canonical class must provide at least:

- A default constructor
- A destructor
- A copy constructor
- An copy assignment operator

Are these elements present and functional?

☒ Yes☐ No

Exercise 01: Towards a more useful fixed-point number class

The previous exercise was a good start, but the class was still pretty useless since it was only able to represent the fixed-point value 0.0.

Makefile

There is a Makefile that compiles using the appropriate flags.

☒ Yes☐ No

Floating-point constructor

Is it possible to construct an instance from a floating-point value?

☒ Yes☐ No

<< operator

Is there a << operator overload and is it functional?

☒ Yes☐ No

Fixed-point value to integer value

A member function "int toInt(void) const;" that converts the fixed-point value to an integer value must be present. Is it functional?

☒ Yes☐ No

Fixed-point value to floating point value

A member function \"float toFloat(void) const;\" that converts the fixed-point value to a float value must be present. Is it functional?

☒ Yes☐ No

Integer constructor

Is it possible to construct an instance from an integer value?

☒ Yes☐ No

Exercise 02: Now we are talking

This exercise adds comparison and arithmetic features to the class.

Makefile

There is a Makefile that compiles using the appropriate flags.

☒ Yes☐ No

Comparison operators

Are the 6 comparison operators (>, =, <=, == and !=) implemented and working properly?

☒ Yes☐ No

Arithmetic operators

Are the 4 arithmetic operators (+, -, * and /) implemented and working properly?

☒ Yes☐ No

Other operators

Are the pre-increment, post-increment, pre-decrement and post-decrement operators implemented and working properly?

☒ Yes☐ No

Static member functions overloads

Last but not least, test the the min() and max() static member functions are implemented and working properly.

☒ Yes☐ No

Exercise 03: BSP

This exercise should have make you realize how easy it is to implement complex algorithms once the basics work as intended.

Makefile

There is a Makefile that compiles using the appropriate flags.

☒ Yes☐ No

Class Point

There is a class Point which has two attributes (x and y) of type Fixed const. It also has a constructor that takes two floats and initialize x and y with those values.

☒ Yes☐ No

Function bsp

There is a function bsp() which prototype is
"bool bsp(Point const a, Point const b, Point const c, Point const point)".
The function returns True if the point is inside the triangle described by
the vertices a, b, and c.
It returns False otherwise.

☒ Yes☐ No

Main and tests

There is at least a main to test that the function bsp() works as required.
Run several test to make sure that the return value is correct.

☒ Yes☐ No

Ratings

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

☒ Ok☐ Outstanding project☐ Empty work☐ Incomplete work☐ Invalid compilation☐ Cheat☐ Crash☐ Concerning situation☐ Leaks☐ Forbidden function

Conclusion

Leave a comment on this evaluation

Finish evaluation

[Terms of use for video surveillance](https://profile.intra.42.fr/legal/terms/1) (https://profile.intra.42.fr/legal/terms/1)

[Declaration on the use of cookies](https://profile.intra.42.fr/legal/terms/2) (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)