



filozoflar

Felsefenin bu kadar ölümcül olacağı nı hiç düşünmemiştim.

Özet: Bu

projede, bir prosesi zincirlemenin temellerini öğreneceksiniz.

Nası l iş parçacı ğı oluşturacağı nı zı göreceksiniz ve muteksleri keşfedeceksiniz.

Versiyon: 10

İçindekiler

I	giriş	2
II	Ortak Talimatlar	3
III	genel bakış	5
IV	genel kurallar	6
V	Zorunlu kı s m	8
VI	Bonus kı smı	9
VII	Sunum ve meslektaş değerlendirmesi	10

Bölüm I

giriş

Felsefe (Yunanca'dan, philosophia, kelimenin tam anlamıyla "bilgelik sevgisi") varoluş, bilgi, değerler, akıl, zihin ve dil hakkındaki genel ve temel soruları incelenmesidir. Bu tür sorular genellikle üzerinde çalışılması veya çözülmesi gereken problemler olarak sorulur. Terim muhtemelen Pisagor tarafından icat edildi (yaklaşık MÖ 570 - 495). Felsefi yöntemler arasında sorgulama, eleştirel tartışma, rasyonel tartışma ve sistematik sunum yer alır.

Klasik felsefi sorular şunları içerir: Herhangi bir şeyi bilmek ve onu kanıtlamak mümkün müdür? En gerçek olan nedir? Filozoflar ayrıca şu gibi daha pratik ve somut sorular sorarlar: Yaşamın en iyi yolu var mı? Adil olmak mı yoksa adaletsiz olmak mı daha iyidir (eğer biri bundan kurtulabilirse)? İnsanlarda özgür irade var mı?

Tarihsel olarak, "felsefe" herhangi bir bilgi bütününe kapsıyordu. Antik Yunan filozofu Aristoteles'in zamanından 19. yüzyıla kadar "doğa felsefesi" astronomi, tıp ve fiziği kapsıyordu. Örneğin, Newton'un 1687 tarihli Doğa Felsefesinin Matematiksel İlkeleri daha sonra bir fizik kitabı olarak sınıflandırıldı.

19. yüzyılda modern araştırma üniversitelerinin büyümesi, akademik felsefenin ve diğer disiplinlerin profesyonelleşmesine ve uzmanlaşmasına yol açtı. Modern çağda, geleneksel olarak felsefenin bir parçası olan bazı araştırmalar, psikoloji, sosyoloji, dilbilim ve ekonomi dahil olmak üzere ayrı akademik disiplinler haline geldi.

Sanat, bilim, politika veya diğer uğraşlarla yakından ilgili diğer araştırmalar felsefenin bir parçası olarak kaldı. Örneğin, güzellik nesnel midir yoksa öznel midir? Birçok bilimsel yöntem var mı yoksa sadece bir tane mi var? Politik ütopya umutlu bir rüya mı yoksa umutsuz bir fantezi mi? Akademik felsefenin başlıca alt alanları arasında metafizik ("gerçeklik ve varlığı'nın temel doğası ile ilgili"), epistemoloji ("bilginin doğası ve temelleri [ve] ... sınırları ve geçerliliği hakkında"), etik, estetik yer alır. , siyaset felsefesi, mantık ve bilim felsefesi.

Bölüm II

Ortak Talimatlar

- Projeniz C dilinde yazılmalıdır.
- Projeniz Norm'a uygun olarak yazılmalıdır. Bonus dosyaları nız/fonksiyonları nız varsa norm kontrolüne dahil edilirler ve içinde norm hatası varsa 0 alırsınız.
- Tanımsız davranışlar dışı fonksiyonları nız beklenmedik bir şekilde (segmentasyon hatası, bus hatası, double free vb.) kapanmamalıdır. Böyle bir durumda, projeniz işlevsiz kabul edilecek ve değerlendirme sırası nda 0 alacaktır.
- Gerekğinde yığıla ayrılan tüm bellek alanı uygun şekilde boşaltılmalıdır. Sızıntı Yok tolere edilecektir.
- Konu gerektiriyorsa, kaynak dosyaları nızı -Wall, -Wextra ve -Werror bayrakları yla istenen çıktı ya derleyecek bir Makefile göndermeli, cc kullanmalı sını z ve Makefile'ı nı z yeniden bağlanmamalıdır.
- Makefile dosyanız en azından \$(NAME), all, clean, fclean ve tekrar.
- Projenize bonusları döndürmek için, Makefile'nize, projenin ana bölümünde yasaklanmış olan tüm çeşitli başlıkları, kitaplıkları veya işlevleri ekleyecek bir kural bonusu eklemelisiniz. Konu başka bir şey belirtmiyorsa, bonuslar farklı bir _bonus.{c/h} dosyası nda olmalıdır. Zorunlu ve bonus kısmı değerlendirme ayrı ayrı yapılır.
- Projeniz libft'inizi kullanmanızı izin veriyorsa, kaynakları nı ve ilişkili Makefile dosyası nı, ilişkili Makefile ile birlikte bir libft klasörüne kopyalamanız gerekir. Projenizin Makefile'ı, kendi Makefile'ı nı kullanarak kitaplığı derlemeli ve ardından projeyi derlemelidir.
- Bu çalışmanın teslim edilmesi gerekmeyecek ve notlandırılmayacak olsa da, projeniz için test programları oluşturmanızı öneririz. Çalışmanızı ve akranlarınızı nı z çalışmaları nı kolayca test etme şansı verecektir. Bu testleri özellikle savunmanız sırası nda faydalı bulacaksınız. Gerçekten de, savunma sırası nda kendi testlerinizi ve/veya değerlendirdiğiniz akranlarınızı nı z testlerini kullanmakta özgürsünüz.
- Çalışmanızı atanmış git deposuna gönderin. Yalnızca git deposundaki çalışmalar derecelendirilecektir. Çalışmanızı notlandırmak için Deepthink atanmışsa, bu yapılacaktır.

filozoflar

Felsefenin bu kadar ölümcül olacağı nı hiç düşünmemiştim.

akran değerlendirmelerinizden sonra. Deepthinkt'in derecelendirmesi sı rası nda çalı şmanı zı n herhangi bir bölümünde bir hata olursa değerlendirme durur.

Bölüm III

genel bakış

Bu görevi başarmak istiyorsanız bilmeniz gerekenler şunlardır:

- Bir veya daha fazla filozof yuvarlak bir masada oturur.
Masanın ortasında büyük bir kase spaghetti var.
- Filozoflar alternatif olarak yer, düşünür veya uyurlar.
Yemek yerken düşünmezler ve uyumazlar; düşünürken ne yemek yiyorlar ne de uyuyorlar; ve tabii ki uyurken ne yemek yiyorlar ne de düşünüyorlar.
- Masada çatal da vardır. Filozof sayısı kadar çatal vardır.
- Spagetti'yi tek çatalla servis etmek ve yemek çok zahmetli olduğu için, bir filozof yemek için sağ ve sol çatalları nı her bir eline birer tane alır.
- Filozof yemeğini bitirince çatalı nı tekrar masaya koyar ve uyumaya başlar. Uyandıktan sonra tekrar düşünmeye başlarlar. Bir filozof açlıktan öldüğünde simülasyon durur.
- Her filozofun yemek yemesi gerekir ve asla aç kalmamalıdır.
- Filozoflar kendi aralarında konuşmazlar.
- Filozoflar başka bir filozofun ölmek üzere olup olmadığını bilmezler.
- Filozofların ölmekten kaçınmaları gerektiğini söylemeye gerek yok!

Bölüm IV

genel kurallar

Zorunlu kı s m için bir program ve bonus kı s m için ayrı bir program yazmalı s nı z (eğer bonus kı smı nı yapmaya karar vererseniz). Her ikisi de aşağı daki kurallara uymak zorundadı r:

- Global değişkenler yasaktı r!
- Program(lar)ı nı z aşağı daki argümanları almalı dı r:
number_of_philosophers time_to_die time_to_eat time_to_sleep
[number_of_times_each_philosopher_must_eat]

filozofları n_sayı s : Filozofları n sayı s ve ayrı ca sayı çatal kümesi.

die_to_time (milisaniye olarak): Bir filozof, ölmek için zamanı son yemeğinin başlangı cı ndan ya da simülasyonun başlangı cı ndan itibaren milisaniyeler içinde yemeye başlamadı ysa, ölürler.

time_to_eat (milisaniye cinsinden): Bir filozofun yemek yemesi için geçen süre. Bu süre zarfı nda iki çatal tutmaları gerekecek.

uyku_zamanı (milisaniye olarak): Bir filozofun uykuda geçireceği süre.

time_of_times_each_philosopher_must_eat (isteğe bağlı argüman): Eğer hepsi filozoflar en az number_of_times_each_philosopher_must_eat kez yemişse, simülasyon durur. Belirtilmezse, bir filozof öldüğünde simülasyon durur.

- Her filozofun 1'den filozofları n_sayı s 'na kadar değişen bir numarası vardı r.
- Filozof sayı s 1, filozof sayı s _filozofları n yanı nda oturuyor. Diğer herhangi bir filozof sayı s N, filozof sayı s N - 1 ile filozof sayı s N + 1 arası nda yer alı r.

Programı nı zı n günlükleri hakkı nda:

- Bir filozofun herhangi bir durum değişikliği aşağı daki gibi biçimlendirilmelidir:

timestamp_in_ms X çatal aldı

timestamp_in_ms X yiyor

timestamp_in_ms X uyuyor

timestamp_in_ms X düşünüyor

timestamp_in_ms X öldü

Timestamp_in_ms'yi milisaniye cinsinden geçerli zaman damgası yla ve X'i filozof numarası yla değiştirin .

- Görüntülenen bir durum mesajı başka bir mesajla karıştı rı lmamalı dı r.
- Bir filozofun öldüğünü bildiren bir mesaj, filozofun gerçek ölümünden en fazla 10 ms sonra gösterilmelidir.
- Yine filozoflar ölmekten kaçınmalı dı r!



Programı nı zda herhangi bir veri yarı şı olmamalı dı r .

Bölüm V

Zorunlu kı s ı m

Program adı	philo
dosyaları teslim et	Makefile, *.h, *.c, philo/ NAME dizininde, all, clean, fclean, re
makefile	number_of_philosophers time_to_die time_to_eat
Argümanlar	time_to_sleep [number_of_times_each_philosopher_must_eat] memset, printf, malloc, ücretsiz, yaz, usleep, gettimeofday, pthread_create, pthread_detach, pthread_join , pthread_mutex_init,
Dı ş işlevler.	pthread_mutex_destroy, pthread_mutex_lock, pthread_mutex_unlock Hayı r
libft yetkili	
Tanı m	Konuları ve muteksleri olan filozoflar

Zorunlu kı s ı m için özel kurallar ş unlardı r:

- Her filozof bir iş parçacı ğ ı olmalı dı r.
- Her filozof çifti arası nda bir çatal vardı r. Bu nedenle, birden fazla filozof varsa, her filozofun sol tarafı nda bir çatal ve sağ tarafı nda bir çatal vardı r. Sadece bir filozof varsa, masada sadece bir çatal olmalı dı r.
- Filozofları n çatalları kopyalaması nı önlemek için çatal durumunu korumalı s ı nı z her biri için bir muteks ile.

Bölüm VI

Bonus kısımları

Program adı	philo_bonus
dosyaları teslim et	Makefile, *.h, *.c, philo_bonus/ NAME dizininde, hepsi, temiz,
makefile	fclean, re number_of_philosophers time_to_die time_to_eat
Argümanlar	time_to_sleep [number_of_times_each_philosopher_must_eat] memset, printf, malloc, ücretsiz, yaz, çatal, öldür, çık, pthread_create, pthread_detach, pthread_join, usleep,
Diğer işlevler.	gettimeofday, waitpid, sem_open, sem_close, sem_post, sem_wait, sem_unlink Hayır
libft yetkili	
Tanımlar	Süreçleri ve semaforları olan filozoflar

Bonus bölümünün programı, zorunlu programla aynı argümanları almalıdır. Global kurallar bölümünün gereksinimlerine uygun olmalıdır.

Bonus kısımları için özel kurallar şunlardır:

- Tüm çatal masanın ortasına konur.
- Bellekte durumları yoktur, ancak kullanılabilir çatal sayısı şu şekilde gösterilir: bir semafor.
- Her filozof bir süreç olmalıdır. Ancak ana süreç bir filozof.



Bonus kısımları, yalnızca zorunlu kısımları MÜKEMMEL ise değerlendirilecektir. Kusursuz demek, zorunlu olan kısımları bütünü olarak yapılmış ve arızasız çalışıyor demektir. TÜM zorunlu gereksinimleri geçmediyseniz, bonus bölümünüz hiç değerlendirilmeyecektir.

Bölüm VII

Başvuru ve akran değerlendirilmesi

Git deponuzdaki ödevinizi her zamanki gibi teslim edin. Savunma sırası anda yalnızca deponuzdaki çalışma değerlendirilecektir. Doğru oldukları ndan emin olmak için dosyaları nı zı n adları nı iki kez kontrol etmekten çekinmeyin.

Zorunlu parça dizini: philo/

Bonus parça dizini: philo_bonus/