

BLG252E NESNE TABANLI PROGRAMLAMA

HW 2.

Son Tarih: 22.04.2019

Bu ödevde, birkaç matematiksel operatörün bir tabloya yerleştirilmesi ve gerçekleştirilmesi istenir. yerleştirilmiş operatörler üzerinde bazı işlemler. Size aşağıdaki Sınıf beyanlarını verilmektedir:

Operatör Sınıfı

**Operator** sınıfı, bir ızgaradaki ( **center\_x** , **center\_y** ) konumunda bulunan genel bir işleci tanımlar. **op\_size** boyutunda ( **op\_size** ögesinin [1..9] aralığında bir sayı olduğunu varsayabilirsiniz). Erişimcisi var Tüm örnek değişkenleri için (getter) ve mutator (setter) yöntemleri. Buna ek olarak, sıfırlamak mümkündür Tüm bu değişkenler **reset** () yöntemiyle birlikte.

Aritmetik İşlemci Sınıfı

**ArithmeticOperator** sınıfı, işleci belirtmek için bir **işaret** alanı ekler. İşaret + , - , x değerlerini alabilir. veya / . Bu küme dışında bir değer kurucuda işaret parametresi olarak verilirse, mesaj

SIGN parametresi geçersiz!

yazdırılmalıdır.

Bir nesnenin somutlaştırılmasından sonra bir **işareti** değiştirmek istemediğimiz için , mutasyona neden olmaz Bunun için yöntem. Ancak, **get\_sign** () accessor yöntemine **sahibiz** .

**print\_operator** () yöntemi şu mesajı yazdırır:

ARITHMETIC\_OPERATÖR [ **imzala** ], CENTER\_LOCATION [ **center\_x** , **center\_y** ], SIZE [ **op\_size** ]

**işareti** operatörü, **center\_x** ve **center\_y** işaretidir nerede operatörün merkezi konuma vardır

(üst sınıf **Operatöründe** tanımlanır ) ve **op\_size** , operatörün boyutudur ( üst **sınıfta** tanımlanır)

**Operatör** )

OperatorGrid Sınıfı

**OperatorGrid** yapıcısı ( **int satırlar** , **int cols** ) , aşağıdakilerden oluşan bir **OperatorGrid** örneği oluşturur **satır** ve **sütun grid\_rows grid\_columns** gösterildiği gibi [Şekil 1'de](#) Not bu **sıra / sütun sayıları 1'den başlar ve 0'dan başlar** .

OperatorGrid (5,8)								OperatorGrid (6,5)				
1	2	3	4	5	6	7	8	1	2	3	4	5
1								1				
2								2				
3								3				
4								4				
5								5				
								6				

Şekil 1: OperatorGrid yapıcısında belirtilen boyutlarda iki ızgara

OperatorGrid **destructor** , dinamik olarak tahsis edilmiş hafıza konumlarını geri vermeli (yani, silmeli)

**ızgara** ve **operatörler** . Aşağıdaki mesajı yazdırır:

YAYINCISI: GERİ DÖN [ **satırlar** , **sütunlar** ] karakterleri.

YAYINCISI : GERİ DÖN [ **num\_operators** ] Operatörleri.

**satırlar grid\_rows** olduğu, **sütunlar grid\_cols** ve **num\_operators num\_operators** örnek değişikendir

Yıkılmakta olan OperatorGrid örneğinin.

**place\_operator** (**ArithmeticOperator \* operator**) yöntem dinamik olarak oluşturulmuş ızgaradaki **ArithmeticOperator** örneği . [Şekil 2](#) , **op\_size** = 1 olan bazı örneklerin yerleşimini göstermektedir. Lütfen, her operatörün şebekeye yerleştirildiğini ve böylece tüm şeklin türünü yansıttığını unutmayın. **sign** örneği değişkenine bakarak belirlenen operatör . Örneğin,

- En soldaki örnek orta konuma [2,2] bir + ( **sign** = '+' ) karakterini yerleştirir ve ekler bir ( **op\_size** = 1 ) + merkezin sağına, soluna, üstüne ve altına karakter.
- En sağdaki örnek orta konuma [4,2] a / ( **sign** = '/' ) karakterini yerleştirir ve ekler bir ( **op\_size** = 1 ) / karakterden merkezin sağ üst ve sol alt **tarafı** .

ArithmeticOperator'ı yerleştirin (5,5)					ArithmeticOperator'ı yerleştirin (5,5)					ArithmeticOperator'ı yerleştirin (5,5)					ArithmeticOperator'ı yerleştirin (5,5)				
1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1	+				1					1					1				
2	+	+			2					2					2				
3	+				3	-	-	-		3	x		x		3			/	
4					4					4		x			4		/		
5					5					5		x		x	5	/			

Şekil 2: Birim operatörlerin (yani op\_size = 1 olan operatörlerin) 5x5 ızgaralara yerleştirilmesi, farklı notasyonlara merkezlenmesi.

Benzer şekilde , [Şekil 3](#) , **op\_size** = 2 olan **ArithmeticOperator** örneklerini merkeze yerleştirilmiş olarak gösterir 5x5 ızgaraları.

ArithmeticOperator'ı yerleştirin (5,5)					ArithmeticOperator'ı yerleştirin (5,5)					ArithmeticOperator'ı yerleştirin (5,5)					ArithmeticOperator'ı yerleştirin (5,5)				
1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1	+				1		x			1	x				1				/
2	+				2			x		2		x			2				/
3	+	+	+	+	3	-	-	-	-	3		x			3			/	
4	+				4			x		4		x			4		/		
5	+				5					5	x			x	5	/			

Şekil 3: Operatörlerin 5x5 ızgaraya op\_size = 2 yerleştirilmesi, operatörler ızgara üzerinde ortalanmıştır (yani, merkez konumu = [3,3]).

Bir operatör yerleştirdiğinizde, operatörün şebekeye uyup uymadığını kontrol etmeniz gerekir. Orada operatörün **en az bir hücre** ızgaranın dışına yerleştirilmiş olan bir **SINIR hatadır** , [Şekil 4](#) gösterir bazı örnek **BORDER ERROR** s (operatör, şebeke dışına taşar).

ArithmeticOperator'ı yerleştirin (5,5)					ArithmeticOperator'ı yerleştirin (3, 4, 2, 5)					ArithmeticOperator'ı yerleştirin (5, 5, 1, '4')						
OperatorGrid de (5,5)					OperatorGrid de (5,5)					OperatorGrid de (5,5)						
1	2	3	4	5	1	2	3	4	5	1	2	3	4	5		
1					1	x			x	1						
2					2		x	x		2			x			
3					3			x		HATA: x ızgaradan çıktı			x			
4	/				4		x		x	4			x			
5	/				5	x			x	5				+		
/ HATA: / şebekeden çıktı															+	+
															+	HATA: + şebekeden çıktı