

Operator Overloadings

Aşağıda UML diagramı verilen kesirli sayı sınıfını yazınız ve verilen main.cpp dosyası ile beraber derleyip çalıştırınız.

Kesirli Sayı UML Diagramı

KesirliSayi
- pay: int - payda: int
+ KesirliSayi() 0 kesirli sayısını oluşturur (0/1)
+ KesirliSayi(pay: int, payda: int) verilen parametrelere göre kesirli sayı oluşturur, oluşturulan kesirli sayıda payda eksi bir sayı olamaz (eski ise gerekli düzenlemeyi yapınız) ve oluşan kesirli sayı olabilecek en sade halde olmalıdır
+ KesirliSayi(tamsayi: int) değeri tamsayi/1 olan kesirli sayıyı oluşturur
- paydaEsitle(ks: KesirliSayi&): void parametre olarak gelen kesirli sayı ile kendisinin paydasını eşitler
- sadelestir(): void kesirli sayıyı olabilecek en sade hale getirir
+ operator=(ks: KesirliSayi): KesirliSayi& parametre olarak gelen kesirli sayının pay ve paydasını kendisine kopyalar
+ operator=(tamsayi: int): KesirliSayi& kendisini parametre olarak gelen tam sayıya eşitler, yani işlem sonucunda kendisi tamsayi/1 değerini alır
+ operator+(ks: KesirliSayi): KesirliSayi kendisiyle parametre olarak gelen kesirli sayıyı toplar, burada dikkat edilmesi gereken işleme giren kesirli sayılar üzerinde hiç bir değişiklik olmamalıdır
+ operator-(ks: KesirliSayi): KesirliSayi kendisinden parametre olarak gelen kesirli sayıyı çıkarır, burada dikkat edilmesi gereken işleme giren kesirli sayılar üzerinde hiç bir değişiklik olmamalıdır

- + operator*(ks: KesirliSayi): KesirliSayi
kendisiyle parametre olarak gelen kesirli sayıyı çarpar, burada dikkat edilmesi gereken işleme giren kesirli sayılar üzerinde hiç bir değişiklik olmamalıdır
- + operator/(ks: KesirliSayi): KesirliSayi
kendisine parametre olarak gelen kesirli sayıyı böler, burada dikkat edilmesi gereken işleme giren kesirli sayılar üzerinde hiç bir değişiklik olmamalıdır
- + operator+=(ks: KesirliSayi): KesirliSayi&
kendisiyle parametre olarak gelen kesirli sayıyı kendisi üzerine toplar
- + operator-=(ks: KesirliSayi): KesirliSayi&
kendisinden parametre olarak gelen kesirli sayıyı çıkarır ve sonucu kendisi üzerine yazar
- + operator*=(ks: KesirliSayi): KesirliSayi&
kendisiyle parametre olarak gelen kesirli sayıyı kendisi üzerinde çarpar
- + operator/=(ks: KesirliSayi): KesirliSayi&
kendisine parametre olarak gelen kesirli sayıyı kendisi üzerinde böler
- + operator++(): KesirliSayi&
ön ek (prefix) artırma işlemi yapar (örn: ++ks), bu işlem kesirli sayıyı bir artırır
- + operator++(i: int): KesirliSayi
son ek (postfix) artırma işlemi yapar (örn: ks++), bu işlem kesirli sayının kendisini bir artırır, fakat artırılmamış halini döner
- + operator==(ks: KesirliSayi): bool
kendisiyle parametre olarak gelen kesirli sayıyı karşılaştırır, bu iki kesirli sayı birbirine eşitse true değilse false döner
- + operator!=(ks: KesirliSayi): bool
kendisiyle parametre olarak gelen kesirli sayıyı karşılaştırır, bu iki kesirli sayı birbirine eşitse değilse true eşitse false döner
- + operator<(ks: KesirliSayi): bool
kendisiyle parametre olarak gelen kesirli sayıyı karşılaştırır, kendisi küçükse true değilse false döner
- + operator>(ks: KesirliSayi): bool
kendisiyle parametre olarak gelen kesirli sayıyı karşılaştırır, kendisi büyükse true değilse false döner
- + operator<=(ks: KesirliSayi): bool
kendisiyle parametre olarak gelen kesirli sayıyı karşılaştırır, kendisi

parametreden küçük eşitse true değilse false döner

- + `operator>=(ks: KesirliSayi): bool`
kendisiyle parametre olarak gelen kesirli sayıyı karşılaştırır, kendisi parametreden büyük eşitse true değilse false döner
- + `<<friend>> operator>>(ist: istream&, ks: KesirliSayi&): istream&`
ist'den iki sayı alır, bunlardan birincisi pay ikincisi paydadır, bu pay ve paydayı kullanarak olabilecek en sade kesirli sayıyı oluşturur (örnek girdi: 4 6, oluşan kesirli sayı 2/3)
- + `<<friend>> operator<<(ost: ostream&, ks: const KesirliSayi&): ostream&`
kesirli sayıyı pay/payda şeklinde ost'ye yazar
- + `<<friend>> test(expected: KesirliSayi, actual: KesirliSayi): bool`
main.cpp içerisinde kodları bulunmakta, tekrar tanımlamayınız, sadece friend function olarak ekleyiniz

Main Fonksiyon Kodu:

main.cpp dosyası ekte verilmiştir.

Örnek Ekran Çıktıları:

Örnek ekran çıktısı için ekteki **cikti.txt** dosyasına bakınız.