**1fd75c3872a94e26ad68c7fa7667bdc82c07dd4ba85f4a6793f7a2b4e943b8fd**

**T.C.**

**SAKARYA ÜNİVERSİTESİ**

**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

PROGRAMLAMA DİLLERİNİN PRENSİPLERİ 4.ÖDEV RAPORU

İŞ PARÇACIKLARI (THREAD) YAPIMI VE YÖNETİMİ

**Grup Elemanları:**

**B151210099 - OĞUZHAN GÜNDOĞDU**

**B151210066 - HASAN AYDOĞMUŞ**

**SAKARYA**

**MAYIS, 2018**

Programlama Dillerinin Prensipleri Dersi

İŞ PARÇACIKLARI (THREAD) YAPIMI VE YÖNETİMİ

OĞUZHAN GÜNDOĞDU[[1]](#footnote-1), HASAN AYDOĞMUŞb

B151210099

B151210066

Özet

Hazırlamış olduğum ödev JAVA programlama dilinde bilgisayarda yapılan belirli bir işin iş parçacıklarına bölünerek nasıl daha hızlı gerçekleştireceğimizi bizlere gösteriyor.Hazırlanan ödev concurrency mantığına dayalı olup multithreading üzerine çalışmaktadır.Bu demek oluyor ki günümüz bilgisayarları her ne kadar güçlü olsa da yapılan işlerde bu denli büyümekte ve her işlemi sırayla yaptırmaktansa her bir adımını bir iş parçacığına bölmek işlemlerimi daha hızlı bir şekilde yapmamıza olanak sağlıyor.(eş zamanlı yapılan işler paralel olarak yapılmakta)

© 2018 Sakarya Üniversitesi.

Bu rapor benim özgün çalışmamdır. Her hangi bir kopya işleminde sorumluluk bana aittir.

Anahtar Kelimeler:JAVA,MULTHREADING,CONCURRENCY,İŞ PARÇACIĞI

1. GELİŞTİRİLEN YAZILIM

* Öncelikle laboratuvar çalışmasına yeterli özeni göstermediğimiz için internetten thread hakkında toplayabildiğimiz kadar bilgi topladık.
* Bunun hemen ardından programda kullanacağımız threadlerin çalıştığı işlem sınıfı Runable interfacesinden implement edilmek suretiyle threadlerimizin iş parçacığı yapısına uygun sınıfımızı oluşturmuş olduk.
* Bu sınıfın içerinde istenen işlemin eş zamanlı(concurrency)olarak yapılabilmesi için string türünde olan çarpan sayımızı oluşturduğumuz algoritmayla paralel olarak iş parçacıkalrının işlemi gerçekleştireceği 4lü parçalara ayrıdık .lakin istek doğrultusunda bu sayıda arttırılabilir.
* İstenilen takdirde bu iş parçacaıklarının sayısını ufak değişikliklerle arttırıp azaltabiliyoruz.bunun ışığında yaptığımız denemeler neticesinde iş parçacığı sayısının çok fazla arttırılması verimi arttırmadığını gözlemledik ve bu sayının bizim gözlemlerimizce ideale yakın olduğunu gördük.
* Sayıyı 4 parça halinde threadlere verdiğimizden 4 thread kullandık.
* Substring kullanarak çarpan olan sayımızın baştan 1. 2. 3. ve 4.basamkların katlarını çarpan sayının basamak adedince parçalara ayırdık
* Ana fonksiyon için programa big integer kütüphanesini import ettik
* Eş zamanlı işlerin çalıştığı sınıftan thread sayısınca nesne ürettik ve bu nesnelere parametre olarak çarpanlarda 1. Ve 2. Ne ek olarak substrğing methodunun alacağı indisleri verdik
* Kullanacağımız threadler için bir havuz oluşturduk ve bu havuz da 4 adet thread atadık
* Ürettiğimiz 4 nesneyi havuza submit ettik,havuz işlemini sonlandırdık
* Eş zamanlı işlemlerin çalışma sürelerini doğru hesaplayabilmek adına while döngüsünde !havuz.isTerminated sorgusunu kullandık.
* Bu arada 4 farklı threadden gelen işlemlerin sonularını birleştirdik
* Karşılaştıramaısnı yapacağımız seri çarpma işleminide ana fonksiyona ekledikve bu işleminde süre ölçümünü yaptık
* Bütün bu işlemleirn sonunda elde ettiğimiz sonuçları dosyaya yazdırdık.

1. ÇIKTILAR

Ödevimiz istenilen doğrultuda ekrana işlemin parallel yahut seri yapılması halinde geçen süreyi ayrı ayrı olmakla birlikte ekrana bastırmaktadır ve bununla birlikte ödev dokümanında belirtilen şekilde dosya da programın içerinde belirlenen (kullanıcıdan alınmayan)sayılar ödevde istenilen şekillerde çarptırılıp sonucu programın içerisinde bulunduğu dosyadaki yaz.txt isimli txt doyasına yazdırılmaktadır.

1. SONUÇ

Günümüz bilgisayarları her ne kadar yakın geçmişe gore güçlü,hızlı,daha performanslı olsa da günümüzde yapılan işlemler kullanılan programlar da aynı derecede daha fazla performans ihtiyacı oluşturuyor.bütün bunların ışığında biz programcı adaylarının daha kaliteli kodlar yazmak suretiyle daha hızlı daha hafif ve azami derecede güvenli programlar geliştirmemiz adına bu çalışma bizlere epey bir yol aldırmıştır.

1. [↑](#footnote-ref-1)