**BLM3003.1 İŞLETİM SİSTEMLERİ**

**2020 GÜZ PROJE-1 RAPOR**

**07.12.2020**

**GRUP ÜYELERİ :**

SÜHENDA HİLAL ETO - 170418016

ELİF GENÇ - 170418038

KAAN İPEK - 171419012

BAHADIR KACAR - 171418001

**PROJENİN GERÇEKLEŞTİĞİ ORTAM :**

Proje, i5-7300HQ (4 CPUs) işlemci, 64 bit işletim sistemi, 16 GB RAM, Monster Abra a5 v12.1 model bilgisayar kullanılarak Virtual Box 6.1 sürümlü sanal makine içerisinde Ubuntu 20.04.1 LTS işletim sistemi kullanılarak yazılmıştır. Sanal makine içerisinde işletim sistemine Ana Bellek’ten 1024 mb ve görüntü belleği 16 mb ayrılmıştır. İşlemci çalıştırma üst sınırı %100 olarak ayarlanmıştır. Bütün bu ayarlar ve sürüm modeli projenin test kısmında çalışma süresini etkileyecektir.

**PROJEDE GERÇEKLEŞTİRİLEN AŞAMALAR :**

Proje, yönergede belirtilen partların tamamını içermektedir(1,2,3,4) .Sıralama algoritması olarak Bubble Sort tercih edilmiştir. Projedeki fonksiyonların işlevleri aşağıda belirtilmiştir.

**FONKSİYONLAR :**

bubbleSort() kullandığımız sıralama fonksiyonu içerisinde swap() fonksiyonunu çağırarak sıralama gerçekleştirilmiştir

child()child processlerin yapacağı işlemler için dosyadan okumak, dosyadan okuduğu verileri sıralamak ve o verileri mid dosyaya yazmak gibi

basamak\_sayisi() dosyadan okuduğumuz verilerin basamak sayılarını bulduğumuz fonksiyon

dosya\_islemleri() girdi dosyalarını ve arayacağımız k. değeri gönderip dosyayı okuyan okunan dosyaları sıralayan ve yazan fonskiyon

ve birçok kütüphaneden hazır fonksiyonlardan yararlanılmıştır

**TEST AŞAMASI :**

Projenin test aşamasında her bir part için 3 adet senaryo belirledik.

**PART-1 TEST**

1. dosyada **k = 1 n=1** text dosyası içinde **8** tane sayı var süre **1090 ms**

1. dosyada **k = 1 n=2** text dosyası içinde **8** tane sayı var, 2.dosyada **6** tane sayı var süre **3542 ms**

1.dosyada **k = 1 n=3** text dosyası içinde **8** tane sayı var , 2.dosyada **6** tane sayı var ,3.dosyada **1000**  tane sayı var süre **6900 ms**

1. dosyada **k = 1 n=4** text dosyası içinde **8** tane sayı var , 2.dosyada **6** tane sayı var 3.dosyada ve 4.dosyada **1000** tane sayı var süre **8587 ms**

**PART-2 TEST**

n =1 dosyada, **8** sayı var süre **658ms** k =1

n =1 dosyada, **8** sayı var süre **681ms** k=2

n =1 dosyada, **8** sayı var süre **693ms** k=2

n =2, 2 dosya var **8** ve **6** sayıdan oluşuyor süre **859ms** k=1

n =2, 2 dosya var **8** ve **6** sayıdan oluşuyor süre **859ms** k=1

n =2, 2 dosya var **8** ve **6** sayıdan oluşuyor süre **926ms** k=2

n = 2, 2dosya var **8** ve **6** sayıdan oluşuyor süre **1162ms** k=3

n = 3, 3 dosya var sırasıyla **8,6,1000** sayıdan oluşuyor süre **3890ms** k=1

n = 3, 3 dosya var sırasıyla **8,6,1000** sayıdan oluşuyor süre **4720ms** k=2

n = 3, 3 dosya var sırasıyla **8,6,1000** sayıdan oluşuyor süre **4435ms** k=3

n = 4, 4 dosya var sırasıyla **8,6,1000,1000** sayıdan oluşuyor süre **8931ms** k=1

n = 4, 4 dosya var sırasıyla **8,6,1000,1000** sayıdan oluşuyor süre **8543ms** k=2

n = 4, 4 dosya var sırasıyla **8,6,1000,1000** sayıdan oluşuyor süre **6449ms** k=3

**PART-3 TEST**

Part-3’teki dosya için thread işlemi olan k = 3 n = 2 dosyada **5** ve **6** adet sayı var süre = **12932ms**

Part-3’teki dosya için thread işlemi olan k = 3 n = 3 dosyada **5,6,1000** adet sayı var süre = **3839ms**

Part-3’teki dosya için thread işlemi olan k = 1 n = 3 dosyada **5,6,100**0 adet sayı var süre = **16848ms**

Part-3’teki dosya için thread işlemi olan k = 2 n = 3 dosyada **5,6,1000** adet sayı var süre = **21512ms**