|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **NUMARA** | **AD-SOYAD** | **İMZA** | **S1** | **S2** | **S3** | **S4** | **T** |
|  |  |  |  |  |  |  |  |

**T.C. C.Ü. MÜH. FAKÜLTESİ BİLGİSAYAR MÜH. BÖLÜMÜ**

**PARALEL PROGRAMLAMA BÜTÜNLEME SINAV SORULARI**

1. Aşağıdaki Java programı paralel olarak rastgele üretilmiş bir string dizisindeki sesli harfleri bulmaktadır. Bu kodun doğru çalışmasını sağlayacak şekilde boşlukları doldurunuz. ( 60 p )

public class charParallel {

static int uzunluk = ...... ;

static public String[] kelimeler = new String[uzunluk];

static public char[] sesliler = new char[8];

static public int[] sayac1, sayac2;

public static void main(String[] args) throws Exception {

sayac1 = new int[8]; sayac2 = new int[8]; int takım = 4;

String str = "aeıioöuü"; sesliler = str.toCharArray();

**//RASTGELE 1000 UZUNLUKLU BİR KELİMELER DİZİSİ ÜRETİN. HER KELİME 100 HARFTEN OLUŞSUN.**

ParallelTeam parTeam = new ParallelTeam(........);

parTeam.execute(new ParallelRegion() {

public void run() {

............................................................................................................................................................................................................................................................................................................................}});

**//KELİMELER DİZİSİNDEN SESLİ HARFLERİ BULUN.**

parTeam = new ParallelTeam(......... );

parTeam.execute(new ParallelRegion() {

public void run() {

................................................................................................................................................................................................................................................................................................................................

............................................................}});

.......... flag = sonucDogruMu();

if (........) {

System.out.println("Doğru " + flag);

} else {

System.out.println("Hata " + flag);

}

}

public static ........ rastgeleKelimeUret(int x) {

Random r = new Random();

String alfabe = "abcdefghıijklmnoöprstuüvxyz";

String kelime = "";

for (int i = 0; i < x; i++) {

kelime += alfabe.charAt(r.nextInt(alfabe.length()));

}

return kelime;

}

public static ......... harfAra(String kelime, char c) {

int count = 0;

for (int i = 0; i < kelime.length(); i++) {

if (kelime.charAt(i) == c) {

count++;

}

}

return count;

}

private static ....... sonucDogruMu() {

boolean fl = true;

**//SESLİ HARFLERİ BULMA İŞLEMİNİ SIRALI YAPIN, SONUÇLARI sayac2 DİZİSİNE ATIN.**

................................................................................................................................................................................................................................................................................................................................

for (int s = 0; s < 8; s++) {

if (sayac1[s] != sayac2[s]) {

fl = false;

}

}

return fl;

}}

1. Speedup and Efficiency kavramlarını (formülleriyle) açıklayınız. ( 10 p )

1. Tarifelemede “Chunk size” kullanmanın avantajı nedir ? . ( 10 p )
2. a. Aşağıdaki problemlerden hangi(si/leri) paralelleştirilemez? Neden? ( 20 p )
3. Paralelleşebilen bir seçeneği OpenMP ile gerçekleyiniz.

1. Seçtiğiniz seçenekteki bir dizinin elemanları toplamını da aynı paralel bölgede bulunuz.

A, B ve C integer dizi olmak üzere;

/\*A,B ***n*** uzunluklu dizi olduğu ve ilklendirilildiği varsayılmıştır. \*/

**#include <omp.h> #include <stdio.h> #include <stdlib.h>**

**#define n 100**

**int main (int argc, char \*argv[]){**

................................................................................................................................................................................................................................................................................................................................................................................................

}