

# ITKP102 Ohjelmointi 1 (6 op), arvosteluraportti

Tentaattori: Antti-Jussi Lakanen

12. huhtikuuta 2023

## Yleistä

Tentti oli pistekeskiarvon 11,4 (keskihajonta 4,5) perusteella tavanomaista vaikeampi. Demohyvitysten kanssa keskiarvo oli 16,3, joten pisterajat asettuivat tällä kertaa kohtalaisen alas. Huomaa, että demopisteet on laskettu tentin päälle, ja arvosana lasketaan vasta sen jälkeen. Opiskelijan omat tehtävät ovat nähtävissä TIMissä alkuperäisellä tenttisivulla. Uusintojen ajankohdat löydät Sisusta.

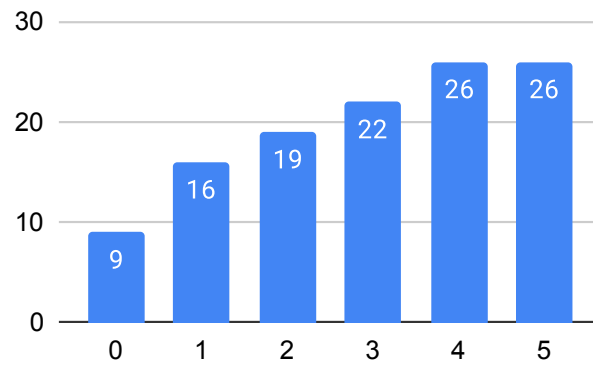
Tehtävä	Teki	Keskiarvo	Automaattinen arvostelu / Tarkastaja
T1	111	2,8	Automaattinen + Oskari Lahtinen
T2	118	4,7	Automaattinen
T3	118	1,9	Automaattinen
T4	118	2,0	Automaattinen
<b>Yht</b>		<b>11,4</b>	(ennen demopisteitä)



JYVÄSKYLÄN YLIOPISTO

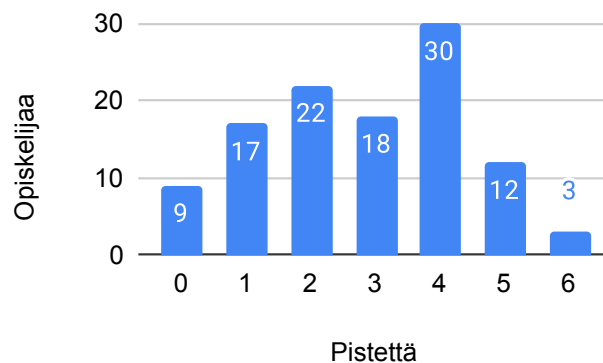
## Arvosteluasteikko ja arvosanajakauma

Arvolause	Pistemäärä (inklusiivinen alaraja)
5	21
4	18
3	15
2	12
1	9



Tentin arvosanajakauma.

## Tehtävä 1 (6 p.)



Tehtävän 1 pistejakauma.

### Kysymys 1.1 (2 p.)

Tee funktio `ParillistenKertoma` käyttäen silmukkaa. Funktio ottaa parametrina kokonaisluvun ja palauttaa sen (mikäli luku on parillinen) ja sitä pienempien parillisten positiivisten kokonaislukujen tulon. Mikäli luku on pariton, vähennetään siitä yksi ennen laskemista.

Esimerkkejä:

- Syötteellä 4 tulos olisi 8, koska  $4 \times 2 = 8$ .
- Syötteellä 8 tulos olisi 384, koska  $8 \times 6 \times 4 \times 2 = 384$ .
- Syötteellä 5 tulos olisi 8, koska  $4 \times 2 = 8$ . (Luku 5 on pariton, joten lasketaan yhtä pienemmän luvun tulos.)

Huomaa, että syötteellä 0 tulos on 1. Sovitaan lisäksi, että syötettäessä negatiivinen kokonaisluku tulos on 0. Mikäli et käytä silmukkaa tässä tehtävässä, voit saada korkeintaan 0.5 pistettä.

### Malliratkaisu

```
/// <summary>
/// Laskee annetun luvun kertoman siten, että vain parilliset
/// luvut lasketaan mukaan. Negatiivisen syötteen tulos on 0.
/// </summary>
/// <param name="n">Luku, josta lähdetään laskemaan kertomaa</param>
/// <returns>Parillisten lukujen kertoma</returns>
public static int ParillistenKertoma(int n)
{
    if (n < 0) return 0;
    int tulos = 1;
    //jos n pariton, vähennetään 1
```

```

    if (n % 2 == 1) n--;
    for (int i = n; i > 0; i -= 2)
    {
        tulos *= i;
    }
    return tulos;
}

```

Arvostelu pohjautui automaattiseen arvioon Liitteessä A Taulukossa 1 esitettyjen arvostelukohteiden/testitapausten kautta. Lopuksi tarkastettiin käsin, onko silmukkaa käytetty ohjeen mukaisesti ja säädettiin pistemäärää sen mukaan. Automaattisilla arvostelukohteilla on tärkeysjärjestys: Mikäli ratkaisu ei toteuta ylempänä olevaa kohdetta, ei sitä alempia arvostelukohteita oteta huomioon. Pisteet pyöristettiin alaspäin lähimpään kokonaislukuun.

## Kysymys 1.2 (1 p.)

Mikä on suurin ParillistenKertoma-funktiolle annettava syöte, jolla tulos mahtuu int-tyypin lukualueeseen?

Vastaus: 19. Syötteellä 19 tulos on 185794560, mutta syötteellä 20 tulos on  $-579076096$ , joten syötteellä 20 tulos ei mahdu int-tyypin lukualueeseen.

## Kysymys 1.3 (2 p.)

Tee funktio ParittomienKertoma käyttäen rekursiota. Funktio ottaa parametrina kokonaisluvun ja palauttaa sen (mikäli luku on pariton) ja sitä pienempien parittomien positiivisten kokonaislukujen tulon. Mikäli luku on parillinen, vähennetään siitä yksi ennen laskemista.

Esimerkkejä:

- Syötteellä 7 tulos on 105, koska  $7 \times 5 \times 3 \times 1 = 105$ .
- Syötteellä 6 tulos on 15, koska  $5 \times 3 \times 1 = 15$ . (Luku 6 on parillinen, joten lasketaan yhtä pienemmän luvun tulos.)
- Huomaa, että syötteellä 0 tulos on 1. Sovitaan lisäksi, että syötettäessä negatiivinen kokonaisluku tulos on 0.

Mikäli et käytä rekursiota tässä tehtävässä, voit saada korkeintaan 0.5 pistettä.

## Malliratkaisu

```

/// <summary>
/// Laskee annetun luvun kertoman siten, että vain parittoman luvut lasketaan mukaan.
/// Mikäli funktiolle annetaan negatiivinen luku, se palauttaa 0.
/// </summary>
/// <param name="n">Luku, josta lähdetään laskemaan kertomaa</param>
/// <returns>Parittomien lukujen kertoma</returns>
public static int ParittomienKertoma(int n)
{

```

```

//jos n on 0 tai 1, lopetetaan rekursio
if (n == 1) return 1;
if (n == 0) return 1;
//jos n on negatiivinen, palautetaan
if (n < 0) return 0;
//jos n on parillinen, vähennetään yksi
if (n % 2 == 0) n--;
return n * ParittomienKertoma(n - 2);
}

```

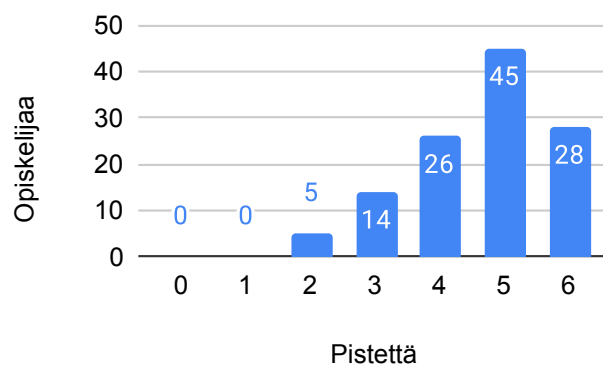
Arvostelu pohjautui automaattiseen arvioon Liitteessä A Taulukossa 2 esitettyjen arvostelukohteiden/testitapausten kautta. Lopuksi tarkastettiin käsin, onko rekursiota käytetty ohjeen mukaisesti ja säädettiin pistemäärää sen mukaan. Arvostelukohteilla on tärkeysjärjestys: Mikäli ratkaisu ei toteuta ylempänä olevaa kohdetta, ei sitä alempia arvostelukohteita oteta huomioon. Pisteet pyöristettiin alaspäin lähimpään kokonaislukuun.

### Kysymys 1.4 (1 p.)

Mikä on suurin ParittomienKertoma-funktiolle annettava syöte, jolla tulos mahtuu int-tyypin lukualueeseen?

Vastaus: 22. Syötteellä 19 tulos on 864408687, mutta syötteellä 23 tulos on  $-1593436679$ , joten syötteellä 23 tulos ei mahdu int-tyypin lukualueeseen.

### Tehtävä 2 (6 p.)



Tehtävän 2 pistejakauma.

- Oletetaan, että tavoitteena on laskea keskiarvo kokonaislukuista, joita on ennalta tuntematon määrä. Mikä seuraavista olisi hyvä esittelyrivi tällaiselle funktiolle?
  - public static int Keskiarvo(int a, int b, int c)
  - public static int[] Keskiarvo(double luvut)
  - public static double Laske(int[] t)

(d) `public static double Keskiarvo(int[] luvut)`

2. Olkoon seuraava katkelma C#-koodia:

```
double[] taulukko = new double[1];  
taulukko[0] = char.MaxValue;
```

Selvitä debuggeria käyttäen taulukko-muuttujan sisältö, kun nämä rivit käännetään ja ajetaan.

- (a) Ohjelma kaatuu ajonaikaiseen virheeseen.
- (b) Taulukossa taulukko on yksi alkio, jonka arvo on 255.
- (c) Taulukossa taulukko on yksi alkio, jonka arvo on 65535.
- (d) Ohjelman kääntäminen epäonnistuu, sillä double-tilaan yritetään sijoittaa char-tyyppinen arvo.

3. Kreikkalainen matemaatikko Eukleides Aleksandrialainen kuvasi n. vuonna 300 eaa. kirjassaan *Alkeet* yksinkertaisen algoritmin jakoyhtälön laskemiseksi. Kulkekaamme Eukleideen jalanjäljissä yhden tenttikysymyksen ajan.

Olkoon seuraava katkelma C#-koodia:

```
public static bool OnkoJaollinen(int jaettava, int jakaja)  
{  
    while (jaettava >= jakaja)  
    {  
        if (jaettava == jakaja) return true;  
        ???  
    }  
    return false;  
}
```

Funktion halutaan palauttavan totuusarvona, onko jaettava tasan jaollinen jakajalla. Esimerkiksi luku 14 olisi jaollinen tasan luvulla 7, koska  $14 = 2 \times 7$ . Luku 15 ei olisi jaollinen luvulla 7, koska  $15 = 2 \times 7 + 1$ .

Kysymysmerkkien tilalta on kuitenkin poistettu rivi koodia. Mikä seuraavista koodiriveistä `**ei toteuta**` yllä kuvattua toiminnallisuutta, jos se laitettaisiin kysymysmerkkien paikalle?

- (a) `return OnkoJaollinen(jaettava - jakaja, jakaja);`
- (b) `return jaettava`
- (c) `jaettava -= jakaja;`
- (d) `jaettava = jaettava`

4. Kertoma on tunnettu siitä, että se kasvaa hyvin nopeasti. Mikä alla olevista luvuista on suurin, joka mahtuu int-tyyppisen kokonaisluvun lukualueelle?

(Mikä on kertoma? Kokonaisluvun  $n$  `**kertoma**` on luvun  $n$  ja kaikkien sitä pienempien positiivisten kokonaislukujen tulo, ja se merkitään  $n!$ . Esimerkiksi  $4! = 4 \times 3 \times 2 \times 1 = 24$ .)

- (a) 10!
- (b) 12!
- (c) 13!
- (d) 20!

5. Mikä seuraavista pitää paikkaansa?

- (a) Taulukosta voi poistaa alkioita.
- (b) taulukko[taulukko.Length] aiheuttaa ajonaikaisen virheen.
- (c) Taulukon sisältö järjestyy automaattisesti nousevaan järjestykseen.
- (d) string[]-tyyppiseen taulukkoon voi sijoittaa alkioiksi char-tyyppisen arvon.

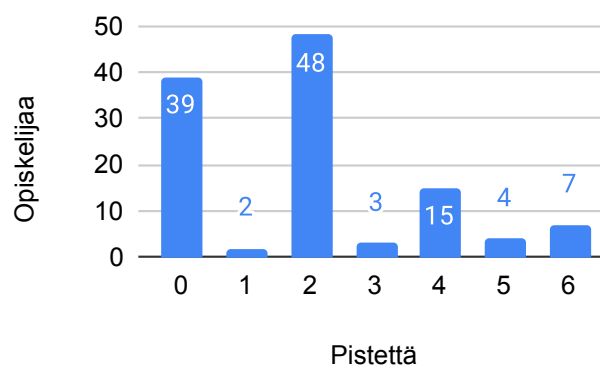
6. Mikä seuraavista pitää paikkaansa?

- (a) Operaattorit `=` ja `==` tarkoittavat samaa asiaa.
- (b) `==` -operaattori palauttaa bool-tyyppisen arvon.
- (c) `==` -operaattoria voi käyttää *\*vain\** if-lauseessa.
- (d) `=` -operaattorilla voidaan vertailla tietyn tyyppisten arvojen yhtäsuuruutta.

## Malliratkaisu

1D 2C 3D 4B 5B 6B

## Tehtävä 3 (6 p.)



Tehtävän 3 pistejakauma.

## Kysymys 3.1 (2 p.)

Tee funktio `SanojenMaara`, joka ottaa parametrina merkkijonon, ja palauttaa kokonaislukuna montako sanaa merkkijonossa on. Välimerkkeinä toimivat välilyönti, pilkku, piste, huutomerkki ja kysymysmerkki. Esimerkiksi syötteellä "Pekka kävi kaupassa" tulos pitäisi olla 3. Toisaalta myös syötteellä "Pekka?? Kävi kaupassa!" tulos pitäisi olla myös 3 siitä huolimatta, että jonossa on useita välimerkkejä peräkkäin.

## Malliratkaisu

```
/// <summary>
/// Merkkijonossa olevien sanojen määrä.
/// Välimerkkeinä ?!., ja välilyönti.
/// </summary>
/// <param name="jono">Merkkijono</param>
/// <returns>Sanojen määrä</returns>
public static int SanojenMaara(string jono)
{
    return jono.Split(new char[] { ' ', '.', '!', '?', ',' },
        StringSplitOptions.RemoveEmptyEntries).Length;
}
```

Arvostelu tehtiin automaattisesti Taulukossa 3 esitettyjen arvostelukohteiden/testitapausten kautta. Arvostelukohteilla on tärkeysjärjestys: Mikäli ratkaisu ei toteuta ylempänä olevaa kohdetta, ei sitä alempia arvostelukohteita oteta huomioon. Pisteet pyöristettiin alaspäin lähimpään kokonaislukuun.

## Kysymys 3.2 (2 p.)

Tee funktio UniikkienSanojenMaara, joka ottaa parametrina merkkijonon, ja laskee kuinka monta eri sanaa ("uniikkia" sanaa) jonossa on. Välimerkkeinä toimivat edellisen kohdan tapaan välilyönti, pilkku, piste, huutomerkki ja kysymysmerkki. Esimerkiksi syötteellä "Pekka kävi kaupassa! Sitten Pekka meni kotiin" tulos pitäisi olla 6, sillä Pekka-sana toistuu kahdesti.

Ratkaisussa ei tarvitse eikä pidä muuttaa kirjainten kokoa, ts. Pekka lasketaan eri sanaksi kuin pekka.

## Malliratkaisu

```
/// <summary>
/// Kuinka monta eri sanaa ("uniikkia" sanaa) jonossa on.
/// </summary>
/// <param name="jono">Merkkijono</param>
/// <returns>Uniikit sanat</returns>
public static int UniikitSanat(string jono)
{
    string[] sanat = jono.Split(new char[] { ' ', '.', '!', '?', ',' },
        StringSplitOptions.RemoveEmptyEntries);
    List<string> uniikit = new List<string>();
    foreach (string sana in sanat)
        if (!uniikit.Contains(sana))
            uniikit.Add(sana);
    return uniikit.Count;
}
```

Arvostelu tehtiin automaattisesti Liitteessä A Taulukossa 4 esitettyjen arvostelukoh- teiden/testitapausten kautta. Arvostelukohteilla on tärkeysjärjestys: Mikäli ratkaisu ei



toteuta ylempänä olevaa kohdetta, ei sitä alempia arvostelukohteita oteta huomioon. Pisteet pyöristettiin alaspäin lähimpään kokonaislukuun.

### Kysymys 3.3 (2 p.)

Tee funktio YleisinSana, joka ottaa parametrina merkkijonon, ja palauttaa sen sanan, joka esiintyy merkkijonossa useimmin. Välimerkkeinä toimivat edellisen kohdan tapaan välilyönti, pilkku, piste, huutomerkki ja kysymysmerkki. Esimerkiksi syötteellä "Pekka, Mari! Paavo, Mari, Irmeli" tulos pitäisi olla Mari, sillä se esiintyy kaksi kertaa, kun muut vain kerran.

Ratkaisussa ei tarvitse eikä pidä muuttaa kirjainten kokoa, ts. Pekka lasketaan eri sanaksi kuin pekka.

#### Malliratkaisu (tapa 1)

```
/// <summary>
/// Palauttaa sen sanan, joka esiintyy merkkijonossa useimmin
/// </summary>
/// <param name="jono">Merkkijono</param>
/// <returns>Useimmin esiintyvä sana</returns>
public static string YleisinSana2(string jono)
{
    // 1. sanat <- splittaa jono
    char[] erotinmerkit = new char[] { ',', ' ', '.', '!', '?' };
    string[] sanat = jono.Split(erotinmerkit,
        StringSplitOptions.RemoveEmptyEntries);

    // 2. järjestä sanat
    Array.Sort(sanat);

    // 3. käy läpi sanat ja pidä kirjaa mikä on suurin perättäisten
    // esiintymien lukumäärä, pidä kirjaa siitä sanasta joka on
    // useimmin esiintyvä (muista ei tarvi kirjanpitoa)
    int maara = 1;
    int maaraNyt = 1;
    string yleisin = sanat[0];

    for (int i = 1; i < sanat.Length; i++)
    {
        if (sanat[i] == sanat[i - 1])
        {
            maaraNyt++;
            if (maaraNyt > maara)
            {
                maara = maaraNyt;
                yleisin = sanat[i];
            }
        }
    }
}
```

```

        else
        {
            maaraNyt = 1;
        }
    }

    // 4. palauta useimmin esiintyvä
    return yleisin;
}

```

## Malliratkaisu (tapa 2)

```

public static string YleisinSana(string jono)
{
    string[] sanat = jono.Split(new char[] { ',', ' ', '.', '!', '?' },
        StringSplitOptions.RemoveEmptyEntries);
    Dictionary<string, int> sanojenMaarat = new Dictionary<string, int>();
    foreach (string sana in sanat)
    {
        if (sanojenMaarat.ContainsKey(sana)) sanojenMaarat[sana]++;
        else sanojenMaarat.Add(sana, 1);
    }

    string yleisin = "";
    int maara = 0;
    foreach (KeyValuePair<string, int> sanaJaMaara in sanojenMaarat)
    {
        if (sanaJaMaara.Value > maara)
        {
            yleisin = sanaJaMaara.Key;
            maara = sanaJaMaara.Value;
        }
    }

    return yleisin;
}

```

Arvostelu tehtiin automaattisesti Liitteessä A Taulukossa 5 esitettyjen arvostelukoh-  
teiden/testitapausten kautta. Arvostelukohteilla on tärkeysjärjestys: Mikäli ratkaisu ei  
toteuta ylempänä olevaa kohdetta, ei sitä alempia arvostelukohteita oteta huomioon.  
Pisteet pyöristettiin alaspäin lähimpään kokonaislukuun.

## Tehtävä 4 (6 p.)

### Kysymys 4.1 (2 p.)

Tee funktio LaskeVokaalit joka ottaa parametrina yhden merkkijonon ja palauttaa merk-  
kijonossa olevien vokaalien lukumäärän. Käytettävät vokaalit ovat aeiouyäö. Niin suur-  
kuin pienaakkoset tulee laskea mukaan.

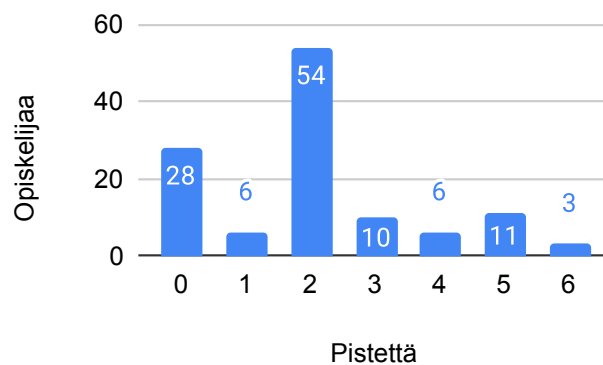
## Malliratkaisu

```
/// <summary>
/// Vokaalien määrä jonossa
/// </summary>
/// <param name="jono">Merkkijono</param>
/// <returns>Vokaalien määrä</returns>
public static int LaskeVokaalit(string jono)
{
    string vokaalit = "aeiouyää";
    int vokaaleja = 0;

    for (int i = 0; i < jono.Length; i++)
    {
        if (vokaalit.Contains(Char.ToLower(jono[i]))) vokaaleja++;
    }

    return vokaaleja;
}
```

Arvostelu tehtiin automaattisesti Liitteessä A Taulukossa 6 esitettyjen arvostelukoh-  
teiden/testitapausten kautta. Arvostelukohteilla on tärkeysjärjestys: Mikäli ratkaisu ei  
toteuta ylempänä olevaa kohdetta, ei sitä alempia arvostelukohteita oteta huomioon.  
Pisteet pyöristettiin alaspäin lähimpään kokonaislukuun.



Tehtävän 4 pistejakauma.

## Kysymys 4.2 (4 p.)

Tee funktio VokaalejaKeskimäärin, joka ottaa parametrina yhden merkkijonon. Funktio palauttaa merkkijonossa olevien sanojen sisältämien vokaalien keskiarvon, kuitenkin vain niiden sanojen osalta jotka sisältävät muitakin kuin pelkkiä vokaaleja.

Esimerkiksi lauseessa "Miia ei tykkää teestä" on kolme sellaista sanaa, jotka sisältävät vokaaleja, mutta eivät pelkkiä vokaaleja (Miia, tykkää, teestä). Sana "ei" sisältää pelkkiä vokaaleja, joten sitä ei lasketa sanojen joukkoon eikä sen vokaaleja lasketa mukaan. Niinpä mukaan laskettavissa sanoissa on yhdeksän vokaalia, joten keskiarvo on  $9/3 = 3$ .

Käytettävät vokaalit ovat aeiouyää. Niin suur- kuin pienenäkköset tulee laskea mukaan.

Sanat tulee erotella välilyöntien kohdalla. Tässä tehtävässä voit olettaa, että syöte sisältää pelkkiä kirjaimia ja välilyöntejä.

### Malliratkaisu

```
/// <summary>
/// Vokaalien määrä jonossa. Käytetään edellisen kohdan vastausta.
/// </summary>
/// <param name="jono">Merkkijono</param>
/// <returns>Vokaalien määrä</returns>
public static int LaskeVokaalit(string jono)
{
    string vokaalit = "aeiouyää";
    int vokaaleja = 0;

    for (int i = 0; i < jono.Length; i++)
    {
        if (vokaalit.Contains(Char.ToLower(jono[i]))) vokaaleja++;
    }

    return vokaaleja;
}

/// <summary>
/// Merkkijonossa olevien sanojen vokaalien keskiarvo,
/// kuitenkin vain niiden sanojen osalta jotka sisältävät
/// muitakin kuin pelkkiä vokaaleja.
/// </summary>
/// <param name="jono">Merkkijono</param>
/// <returns>Vokaalien keskiarvo</returns>
public static double VokaalejaKeskimäärin(string jono)
{
    string[] pilkottu = jono.Split(" ");

    int summa = 0;
    int sanoja = 0;

    for (int i = 0; i < pilkottu.Length; i++)
    {
        int vokaalit = LaskeVokaalit(pilkottu[i]);
        if (vokaalit < pilkottu[i].Length)
        {
            summa += vokaalit;
            sanoja++;
        }
    }
}
```

```
    if (sanoja <= 0) return 0.0;  
    return 1.0 * summa / sanoja;  
}
```

Arvostelu tehtiin automaattisesti Liitteessä A Taulukossa 7 esitettyjen arvostelukoh-  
teiden/testitapausten kautta. Arvostelukohteilla on tärkeysjärjestys: Mikäli ratkaisu ei  
toteuta ylempanä olevaa kohdetta, ei sitä alempia arvostelukohteita oteta huomioon.  
Pisteet pyöristettiin alaspäin lähimpään kokonaislukuun.

## **A Arviointikohteet**

Syöte	Tulos	Selitys	Virhe	Pistemäärä
6	48	Funktio toimii pienellä parillisella luvulla	Funktio ei toimi oikein pienellä parillisella luvulla.	0.75
5	8	Funktio toimii parittomalla luvulla kutsuttaessa.	Funktio ei toimi oikein parittomalla luvulla kutsuttaessa.	0.75
0	1	Funktio toimii nollalla.	Funktio ei toimi oikein nollalla kutsuttaessa.	0.25
-7	0	Funktio toimii negatiivisella luvulla kutsuttaessa.	Funktio ei toimi oikein negatiivisella luvulla kutsuttaessa.	0.25

Taulukko 1: Kysymyksen 1.1 arviointikohteet

Syöte	Tulos	Selitys	Virhe	Pistemäärä
7	105	Funktio toimii pienellä parittomalla luvulla	Funktio ei toimi oikein pienellä parittomalla luvulla.	0.75
6	15	Funktio toimii parillisella luvulla kutsuttaessa.	Funktio ei toimi oikein parillisella luvulla kutsuttaessa.	0.75
0	1	Funktio toimii nollalla.	Funktio ei toimi oikein nollalla kutsuttaessa.	0.25
-7	0	Funktio toimii negatiivisella luvulla kutsuttaessa.	Funktio ei toimi oikein negatiivisella luvulla kutsuttaessa.	0.25

Taulukko 2: Kysymyksen 1.3 arviointikohteet

Syöte	Tulos	Selitys	Virhe	Pistemäärä
“Pekka kävi kaupassa”	3	Kolme sanaa välilyönnillä eroteltuna.	Välilyönneillä erotellut sanat tulisi laskea eri sanoiksi.	0.5
“Pekka kävi kaupassa”	3	Kolme sanaa useammalla välilyönnillä eroteltuna.	Vaikka välimerkkejä on perättäin useampi kuin yksi, se ei pitäisi vaikuttaa sanojen lukumäärään.	0.5
“Pekka kävi kaupassa, ja osti karkkia! Sitten hän meni kotiin.”	10	Sanojen välissä on useita välimerkkejä.	Vaikka välimerkkejä on perättäin useampi kuin yksi, se ei pitäisi vaikuttaa sanojen lukumäärään.	0.5
“”	0	Tyhjässä jonossa ei ole yhtään sanaa.	Tyhjässä jonossa ei ole yhtään sanaa.	0.5

Taulukko 3: Kysymyksen 3.1 arviointikohteet



Syöte	Tulos	Selitys	Virhe	Pistemäärä
“Pekka kävi kaupassa”	3	Kaikki sanat erilaisia	Välilyönneillä erotellut sanat tulisi laskea eri sanoiksi.	0.5
“Pekka kävi kaupassa! Sit-ten Pekka meni kotiin.”	6	Pekka toistuu kahdesti. Useita välimerkkejä peräkkäin	Pekka-sana esiintyy kahdesti. Myös, vaikka välimerkkejä on perättäin useampi kuin yksi, se ei pitäisi vaiku ttaa sanojen lukumäärään.	0.25
“Pekka Pekka Pekka”	1	Kaikki sanat samoja	Sama sana pitäisi laskea vain ker- ran.	0.25
“Pekka Pekka Pekka Kari Kari Sanna”	3	Useita sanoja, useampia samoja sanoja	Samat sanat pitäisi laskea vain kertaalleen.	0.25
“Pekka Kari Pekka Kari Sanna Pekka”	3	Useita samoja sanoja, jotka eivät ole peräkkäin.	Samat sanat pitäisi laskea vain kertaalleen.	0.25
“ !?,. ”	0	Jono sisältää pelkkiä välimerkkejä.	Pelkkiä välimerkkejä sisältävä jono ei sisällä yhtään sanaa.	0.25
“”	0	Tyhjä merkkijono	Tyhjässä jonossa ei ole yhtään sanaa.	0.25

Taulukko 4: Kysymyksen 3.2 arviointikohteet

Syöte	Tulos	Selitys	Virhe	Pistemäärä
“maito juusto, juusto voi kurkku juus- to, keksi? kahvi juusto pinaatti tofu tofu”		Useita sanoja.	Funktion tulisi palauttaa useim- min esiintyvä sana.	0.25
“salaatti salaatti salaatti salaatti ? salaatti salaatti tofu voi voi tofu maito voi tofu tofu maito ? lei- pä maito ? voi leipä ! tofu ”		Satunnainen syöte.	Funktio ei toimi satunnaisella syötteellä.	0.5
“Pekka kävi kau- passa”	Pekka	Kaikki sanat esiintyvät yhtä monta kertaa.	Yhtä monta kertaa esiintyvistä sa- noista tulee valita ensimmäinen.	0.25
“maito maito maito maito leipä lei- pä voi voi voi salaatti salaatti salaatti”	maito	Useat sanat esiintyvät yhtä monta kertaa.	Yhtä monta kertaa esiintyvistä sa- noista tulee valita ensimmäinen.	0.25
“”		Tyhjä syöte, tyhjä tulos.	Tyhjällä syötteellä tulos pitäisi olla tyhjä merkkijono.	0.25
“maito, maito!, maito maito leipä... leipä voi voi voi salaatti salaatti salaatti”	maito	Useat sanat esiintyvät yhtä monta kertaa.	Yhtä monta kertaa esiintyvistä sa- noista tulee valita ensimmäinen.	0.25
“ !?,.”		Syöte sisältää pelkkiä erikoismerk- kejä.	Pelkkiä välimerkkejä sisältävä jo- no ei sisällä yhtään sanaa, tulos on tyhjä jono.	0.25

Taulukko 5: Kysymyksen 3.3 arviointikohteet

Syöte	Tulos	Selitys	Virhe	Pistemäärä
“aapinen”	4	Osaa etsiä vokaaleja yksittäisestä sanasta	Väärä määrä vokaaleja	0.5
“ooooo”	5	Kaikki merkit samoja vokaaleja	Jokainen vokaali tulee laskea	0.5
“bbbbb”	0	Kaikki merkit samoja konsonantteja	Merkkijonossa ei ole yhtään vokaalia	0.25
“Aku Ankka”	4	Ottaa huomioon myös isot vokaalit	Ei ota huomioon isoja vokaaleja	0.25
“”	0	Tyhjä merkkijono	Tyhjässä jonossa ei ole yhtään vokaalia.	0.25
“äidin ääni ärsytti opiskelijaa, joka yritti ymmärtää öljy- ja ydin-teollisuutta”	35	Sisältää jokaisen vaaditun vokaalin vähintään kerran	Ei ota huomioon kaikkia vokaaleja tai välilyöntejä	0.25

Taulukko 6: Kysymyksen 4.1 arviointikohteet

Syöte	Tulos	Selitys	Virhe	Pistemäärä
“äiti tykkää tees- tä”	3	Laskee keskiarvon oikein jos ei ole yhtään sanaa, jossa on pelkkiä vokaaleja.	Ei laske sanojen sisältämien vokaalien keskiarvoa oikein.	1
“Isä tarjoaa Miialle teetä”	3.25	Ottaa huomioon suuraakkoset.	Ei ota huomioon suuraakkosia.	1
“Miia ei tykkää teestä”	3	Ottaa huomioon sanat, joissa on pelkkiä vokaaleja.	Ei ota huomioon pelkkiä vokaaleja sisältäviä sanoja.	1
“aeiouyäö”	0	Syöte sisältää pelkkiä vokaaleja.	Ei ota huomioon syötettä jossa on pelkkiä vokaaleja.	0.5
“”	0	Tyhjä merkkijono	Ei ota huomioon tyhjää merkkijonoa	0.25
“bbb ccc ddd”	0	Merkkijonossa pelkkiä vokaaleja	Ei ota huomioon merkkijonoa, jossa on pelkkiä vokaaleja	0.25

Taulukko 7: Kysymyksen 4.2 arviointikohteet