

Statistička analiza podataka - otpornici

Student: Vladimir Bilogrević

JMBAG: 0135267588

Smjer: Komunikacijska i računalna tehnika

Sadržaj

Teorijski uvod	3
Opis i ideja	3
Raspisano rješenje zadatka te definirane ulazne i izlazne veličine koje se javljaju u programu	3
Kod programa	5
Zaključak	8
Literatura	9

Teorijski uvod

Svrha ovog problemskog zadatka je točnije i brže računanje statističke analize otpornika, problemskog zadatka iz kolegija Mjerenja u elektrotehnici. Program će korisniku omogućiti unos nazivne vrijednosti otpornika, broj ponavljanja mjerenja te izmjerene vrijednosti otpornika. Vrijednosti otpornika ne mogu biti negativne. Program će potom izračunati i ispisati statističku analizu koja se sastoji od aritmetičke sredine, apsolutne pogreške, relativne pogreške, postotne pogreške, standardne devijacije pojedinačnih mjerenja, relativno standardno odstupanje pojedinačnih mjerenja (koeficijent varijacije) te relativno standardno odstupanje pojedinačnih mjerenja (postotak).

Opis i ideja

Cilj ovog programa je automatizirati računanje statističke analize otpornika.

Nakon mjerenja vrijednosti otpornika, program će od korisnika tražiti unos nazivne vrijednosti otpornika, zatim će korisnik unijeti n broj ponovljenih mjerenja te će na kraju unijeti dobivene mjerne rezultate.

Nakon unosa poznatih veličina i broja ponovljenih mjerenja, program će sam izračunati te ispisati na zaslon i u datoteku rezultati.txt dobivene vrijednosti statističke analize.

Ovaj problemski zadatak je namijenjen ukoliko imamo naručeni poveći broj otpornika koji će biti ugrađeni u neki elektrotehnički proizvod. Potrebno je ispitati uzorak, što je uzorak veći, to je veća pouzdanost da je reprezentativan. Da bi znali da li je smijemo zaprimiti i ugraditi naručene otpornike, potrebno je provesti statističku analizu podataka. Otpornici će biti zaprimljeni zadovoljavaju li odgovarajuće kriterije.

Raspisano rješenje zadatka te definirane ulazne i izlazne veličine koje se javljaju u programu

Za opis cjelokupnog postupka rješavanja problemskog zadatka statističke analize, koristiti ćemo primjer s auditornih vježbi.

ZADATAK:

1. U laboratoriju proizvođača elektroničkih komponenta provjerava se vrijednost otpornika **nazivne vrijednosti** od **1 Ω**. Kako bi se umanjio utjecaj slučajnih pogrešaka, mjerenje omotrom ponavlja se **5 puta** u istim uvjetima. Dobiveni su sljedeći **mjerni** rezultati: 995 mΩ; 997 mΩ; 1000 mΩ; 1001 mΩ; 1004 mΩ. Izračunajte:
- aritmetičku sredinu otpora pojedinačnih mjerenja
 - apsolutnu pogrešku mjerenja
 - relativnu pogrešku mjerenja
 - postotnu pogrešku mjerenja
 - standardnu devijaciju pojedinačnih mjerenja
 - relativno standardno odstupanje pojedinačnih mjerenja (koeficijent varijacije)
 - relativno standardno odstupanje pojedinačnih mjerenja izraženo u postotcima
- Sve veličine koje imaju mjernu jedinicu iskažite u **miliomima** i **omima**.

Postupak rješavanja:

Korisnik treba unijeti nazivnu vrijednost (R_n) u Ω, zatim treba unijeti n broj ponovljenih mjerenja te zatim unosi izmjerene vrijednosti (R_i) u mΩ.

Definiranje ulaznih veličina

R_n [Ω]	Pozitivan realni broj
n	Pozitivan cijeli broj
R_i [mΩ]	Pozitivan realni broj

Aritmetička sredina (\bar{R}) će se izračunati uz pomoć formule $\bar{R} = \frac{\sum_{i=1}^n R_i}{n}$, ima mjernu jedinicu.

Apsolutnu pogrešku mjerenja (p_a) ćemo dobiti uz pomoć aritmetičke sredine i nazivne vrijednosti otpora, no prije toga će *mjera* (R_p) poprimiti vrijednost \bar{R} ($R_p = \bar{R}$), formula za računanje apsolutne pogreške $p_a = R_n - R_p$, ima mjernu jedinicu.

Relativnu pogrešku mjerenja (p) ćemo dobiti kao omjer apsolutne pogreške i mjere, $p = \frac{p_a}{R_p}$,

nema mjernu jedinicu, već se iskazuje kao broj.

Postotnu pogrešku mjerenja ($p\%$) ćemo dobiti kao postotak relativne pogreške mjerenja(p),

$p\% = p \cdot 100\%$, iskazuje se u postotcima

Standardnu devijaciju pojedinačnih mjerenja (s) ćemo dobiti uz pomoć izmjerenih vrijednosti (R_i), aritmetičke sredine(\bar{R}) te n broja ponovljenih mjerenja umanjjenih za jedan, ima mjernu jedinicu.

$$s = \sqrt{\frac{\sum_{i=1}^n (R_i - \bar{R})^2}{n - 1}}$$

Relativno standardno odstupanje pojedinačnih mjerenja (V_c) ćemo dobiti kao omjer standardnog odstupanja pojedinačnih očitavanja i aritmetičke sredine, iskazuje se kao broj.

$$V_c = \frac{s}{\bar{R}}$$

Relativno standardno odstupanje pojedinačnih mjerenja izraženo u postocima (s%) ćemo dobiti kao postotak relativnog standardnog odstupanja pojedinačnih mjerenja (V_c), iskazuje se u postotku.

$$s\% = V_c \cdot 100\%$$

Definiranje izlaznih veličina

R- [mΩ / Ω]	Pozitivan realni broj
pa [mΩ / Ω]	Pozitivan realni broj
p	Pozitivan realni broj
p% [%]	Pozitivan realni broj
s [mΩ / Ω]	Pozitivan realni broj
vc	Pozitivan realni broj
s% [%]	Pozitivan realni broj

Kod programa

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

#define MAX 100

void Ucitaj(float *pr, int nn); // Prototip funkcije za učitavanje izmjerenih rezultata
float Aritmeticka(float *pr, int nn); // Prototip funkcije za racunanje aritmeticke sredine
float Standarna(float *pr, int nn, float arit); // Prototip funkcije za racunanje standardne devijacije
void Ispisdat(FILE *pOpen, float arit, float apso, float rela, float post, float stand, float rasopm, float rasop2); // Funkcija za ispit u datoteku

int main(void)
{
    // Definicija varijabli
    int n;
    float rn, arit, apso, rela, post, stand, rsopm, rsop2;

    // Definicija polja i inicijalizacija pokazivaca na prvu adresu polja
    float r[MAX];
    float *pr = &r[0];

    FILE *pOpen = fopen("rezultati.txt", "w");
    if(pOpen == NULL)
    {
        printf("Greska pri otvaranju!\n");
        exit(1);
    }

    // Komunikacija s korisnikom i unos nazivne vrijednosti otpornika
    do{
        printf("Nazivna vrijednost otpornika (Rn) u Ohmima: ");
```

```

scanf("%f", &rn);
if(rn <= 0)
    printf("Vrijednost ne moze biti 0 ili negativan broj\n");
}while(rn <= 0);

printf("\nBroj mjerenja je ogranicen na 100!\n");

// Komunikacija s korisnikom i unos broja mjerenja
do{
    printf("Broj mjerenja (n): ");
    scanf("%d", &n);
    if(n > MAX || n < 1)
        printf("Unijeli ste neogovarajuci broj mjerenja, ponovite!\n");
}while(n > MAX || n < 1);

Ucitaj(&r[0], n); // Poziv funkcija za ucitavanje izmjerenih vrijednosti
arit = Aritmeticka(&r[0], n); // Funkcija za racunanje aritmeticke sredine
apso = fabs(rn - arit); // Racunanje apsolutne pogreske mjerenja
rela = apso / rn; // Racunanje relativne pogreske mjerenja
post = rela * 100; // Racunanje postotne pogreske mjerenja
stand = Standarna(&r[0], n, arit); // Varijabli stand se prirodaje vrijednost izracunata funkcijom standardne devijacije
rsopm = stand / arit; // Racunanje relativno standardno odstupanje pojedinacnih mjerenja (koeficijent)
rsop2 = rsopm * 100; // Racunanje relativno standardno odstupanje pojedinacnih mjerenja (postotak)

printf("\n");

// Komunikacija s korisnikom i ispis dobivenih vrijednosti na zaslon
printf("R= %.2f mOhm = %.4f ohm\n", arit*1000, arit);
printf("pa= %.2f mOhm = %.4f ohm\n", apso*1000, apso);
printf("p= %.4f\n", rela);
printf("p%= %.2f %%\n", post); // Ispis u datoteku
printf("s= %.2f mOhm = %.7f\n", stand*1000, stand);
printf("vc= %.5f\n", rsopm);
printf("s%= %.3f %%\n", rsop2);

Ispisdat(pOpen, arit, apso, rela, post, stand, rsopm, rsop2); // Poziv funkcije za ispis u datoteku
fclose(pOpen);

return 0;
}

// Funkcija za ucitavanje izmjerenih vrijednosti
void Ucitaj(float *pr, int nn)
{
    int i;
    printf("\nUnesite mjerne rezultate (Ri) u mOhmima\n");
    for(i = 0; i < nn; i++)
    {
        printf("%d. element: ", i+1);
        scanf("%f", pr + i);
        *(pr+i) = *(pr+i) / 1000;
        while(*(pr+i) < 0)
        {
            printf("Pogresan unos ");
            scanf("%f", pr+i);
            *(pr+i) = *(pr+i) / 1000;
        }
    }
}

```

```

    }
}

// Funkcija za racunanje aritmeticke sredine
float Aritmeticka(float *pr, int nn)
{
    float suma = 0;
    int i;
    for(i = 0; i < nn; i++)
        suma += *(pr+i);
    return suma / nn;
}

// Funkcija za racunanje standardne devijacije
float Standarna(float *pr, int nn, float arit)
{
    float suma = 0;
    int i;
    for(i = 0; i < nn; i++)
    {
        suma += pow(*(pr+i) - arit, 2);
    }
    float ss = sqrt(suma/(nn-1));
    return ss;
}

// Funkcija za ispit u datoteku
void Ispisdat(FILE *pOpen, float arit, float apso, float rela, float post, float stand, float rsopm, float rsop2)
{
    fprintf(pOpen, "R-= %.2f mOhm = %.4f ohm\n", arit*1000, arit); // Ispis u datoteku
    fprintf(pOpen, "pa= %.2f mOhm = %.4f ohm\n", apso*1000, apso); // Ispis u datoteku
    fprintf(pOpen, "p= %.4f\n", rela); // Ispis u datoteku
    fprintf(pOpen, "p%= %.2f %%\n", post); // Ispis u datoteku
    fprintf(pOpen, "s= %.2f mOhm = %.7f\n", stand*1000, stand); // Ispis u datoteku
    fprintf(pOpen, "vc= %.5f\n", rsopm); // Ispis u datoteku
    fprintf(pOpen, "s%= %.3f %%\n", rsop2); // Ispis u datoteku
}

```

Prikaz rješenja na zaslonu računala

Prikaz na zaslonu

```
D:\Faks\Programiranje\lab4\lab4\bin\Debug\lab4.exe
Nazivna vrijednost otpornika (Rn) u Ohmima: 1
Broj mjerenja je ogranicen na 100!
Broj mjerenja (n): 5
Unesite mjerne rezultate (Ri) u mOhmima
1. element: 995
2. element: 997
3. element: 1000
4. element: 1001
5. element: 1004
R-= 999.40 mOhm = 0.9994 ohm
pa= 0.60 mOhm = 0.0006 ohm
p= 0.0006
p= 0.06 %
s= 3.51 mOhm = 0.0035071
vc= 0.00351
s= 0.351 %
Process returned 0 (0x0)   execution time : 10.675 s
Press any key to continue.
```

Prikaz u datoteci

```
rezultati - Blok za pisanje
Datoteka  Uređivanje  Oblikovanje  Prikaz  Pomoć
R-= 999.40 mOhm = 0.9994 ohm
pa= 0.60 mOhm = 0.0006 ohm
p= 0.0006
p= 0.06 %
s= 3.51 mOhm = 0.0035071
vc= 0.00351
s= 0.351 %
```

Moguće greške pri upisu

<pre>D:\Faks\Programiranje\lab4\lab4\bin\Debug\lab4.exe Nazivna vrijednost otpornika (Rn) u Ohmima: -5 Vrijednost ne može biti manja od 0 ili negativan broj Nazivna vrijednost otpornika (Rn) u Ohmima: 0 Vrijednost ne može biti manja od 0 ili negativan broj Nazivna vrijednost otpornika (Rn) u Ohmima: 5 Broj mjerenja je ogranicen na 100! Broj mjerenja (n): 101 Unijeli ste neogovarajuci broj mjerenja, ponovite! Broj mjerenja (n): -101 Unijeli ste neogovarajuci broj mjerenja, ponovite! Broj mjerenja (n): 0 Unijeli ste neogovarajuci broj mjerenja, ponovite! Broj mjerenja (n): 5 Unesite mjerne rezultate (Ri) u mOhmima 1. element: 900 2. element: -500 Pogresan unos 500 3. element: 909 4. element: 10000 5. element: 100</pre>	<p>Korisnik ne može unijeti negativni broj ili 0</p> <p>Korisnik ne može unijeti broj veći od 100, negativni broj ili 0</p> <p>Korisnik ne može unijeti negativni broj</p>
--	---

Zaključak

Uz pomoć ovog programskog rješenja, možemo puno brže, lakše i točnije odrediti statičku analizu, također ovaj program može uveliko olakšati profesorima ispravljanje kolokvijima i ispita jer se ovaj zadatak učestalo pojavljuje. U ovome programskom rješenju smo obuhvatili sve bitne stvari koje smo učili na kolegiju, a to su polja, pokazivači, funkcije i rad s datotekama. Ukoliko korisnik treba unositi vrijednosti otpora s drugačijim prefiksom, to može učiniti uz male preinake u kodu.

Literatura

1. Bednjanec, Andrea: Predavanje iz kolegija Mjerenja u elektrotehnici – „MUE_2_predavanje_Pogreske mjerenja, granice pogresaka, statisticka obrada_2022_2023“, preuzeto sa stranice moj.tvz.hr
2. Bednjanec, Andrea: Auditorne vježbe iz kolegija Mjerenja u elektrotehnici – „MUE_1_AV_Statisticka analiza podataka I dio“, preuzeto sa stranice moj.tvz.hr
3. Zuppa Bakša, Vatroslav: Predavanja iz kolegija Programiranje, Tehničko veleučilište u Zagrebu (2024)