

Praktikum 2 Abgabe bis 06.12.2020

Name, Vorname: Cihat,Özkurt

GMID: inf2323

Betreuer im Praktikum: Gross, Julian

Mat.-Nr.: 11148632

Datum: 06.12.2020

Aufgabe 1: Programmieren in C

In AP 1 wurde die Klausur geschrieben. Prof. Dr. Victor hat sie bewertet und muss nun noch den Notenspiegel, die Durchschnittsnote und die Durchfallquote ausgeben. Durchgefallen ist, wer eine 5 bekommen hat. (Es gibt nur die Noten 1 bis 5).

Schreiben Sie ein C-Programm, das per Benutzereingabe die Anzahl der jeweiligen Noten in ein Array speichert. Anschließend sollen die Werte aus dem Array gelesen und formatiert in einer Tabelle ausgegeben werden. Außerdem wird die Durchschnittsnote und die Durchfallquote in Prozent berechnet.

Hinweis: Benutzen Sie für die Ein- sowie die Ausgabe jeweils eine Schleife.

Die Ausgabe des Programms sollte in etwa so aussehen:

Programm Notenspiegel

Bitte geben Sie die Anzahl der Arbeiten mit der Note 1 ein: ____

Bitte geben Sie die Anzahl der Arbeiten mit der Note 2 ein: ____

.
. .
.

Bitte geben Sie die Anzahl der Arbeiten mit der Note 5 ein: ____

Der Notenspiegel sieht dann so aus:

Note	1	2	3	4	5	Gesamt
Anzahl						

Durchschnittsnote: 2.9 (als Beispielwert)

Durchfallquote: 52.5 % (als Beispielwert)

Lösung:

<Kopieren Sie bitte Ihr Programm hier hin. >

```

#include<stdio.h>
int main(){

    int a[5];
    double durchfallquote,durchschnittsnote,personenZahl=0,summe = 0;

    //Eingabe der Noten
    for(int i=0;i<5;i++){
        printf("\nBitte geben Sie die Anzahl der Arbeiten mit der Note %i ein:",i+1);
        scanf("%i",&a[i]);
        summe = summe + (i+1)*a[i];
        personenZahl = personenZahl+a[i];
    }

    //Ausgabe der Noten
    printf("\n");
    printf("\nNote\t1\t2\t3\t4\t5\nGesamt");
    for(int i=0;i<5;i++)
        printf("\t%i",a[i]);

    durchschnittsnote = summe/personenZahl;
    durchfallquote = a[4]*100.f/personenZahl;

    //Ergebnisse des Programm
    printf("\n");
    printf("\ndurchschnittsnote %f\n",durchschnittsnote);
    printf("\ndurchfallquote %f%\n",durchfallquote);

    return 0;
}

```

Aufgabe 2: Programmieren in C

Schreiben Sie ein C-Programm, das die Primzahlen zwischen 2 bis maximal 1000 berechnet und zwar nach der altherwürdigen Methode „Sieb des Eratosthenes“ (3. Jahrhundert. v. Chr.): Das Verfahren berechnet für eine eingegebene natürliche Zahl (>1) alle Primzahlen bis einschließlich dieser Zahl. Für falsche oder gar keine Eingaben geben Sie bitte eine entsprechende Fehlermeldung aus.

Hinweis: Verwenden Sie ein `int`-Array, dessen Komponenten 0 oder 1 sein können. Das Ziel besteht darin, dem i -ten Element den Wert 1 zu zuweisen, falls i eine Primzahl ist, und andernfalls den Wert 0. Dies wird erreicht, indem für jedes i alle Elemente des Arrays, die einem beliebigen Vielfachen von i entsprechen, auf 0 gesetzt werden. Danach wird das Feld nochmals durchlaufen, um alle gefundenen Primzahlen auszugeben.

Lösung:

<Kopieren Sie bitte Ihr Programm hier hin. >

```

#include<stdio.h>
int main(){
    int a=0,array[1000],primzahlen[1000],kontrolle;

    //Eingabe der Zahl
    printf("Bitte geben Sie ein Zahl zwischen 2-1000: ");
    scanf("%i",&a);

    //Kontrolle von Zahl ob sie zwischen 2-1000 ist
    if(a>2 && a<1000){
        int summe=0;
        //Überprüfung der Zahlen ob sie sind Primzahl ist. Wenn eine Zahl Primzahl ist Zahl->1,
        nicht Primzahl ist Zahl->0
        //array[] ist für 0 oder 1
        //primzahlen[] ist für alle primzahlen bis eingegebene Zahl
        for(int i=2;i<a;i++){
            array[i]=0;
            kontrolle=0;
            for(int j=2;j<i;j++){
                if(i%j==0)
                    kontrolle=1;
            }
            if(kontrolle==0){
                array[i]=1;
                primzahlen[summe]=i;
                summe++;
                printf("%i -> %i \n",i,array[i]);
            }
            else{
                array[i]=0;
                printf("%i -> %i \n",i,array[i]);
            }
        }
        printf("Primzahlen \n");
        for(int i=0;i<summe;i++)
            printf("%i\n",primzahlen[i]);

    }else
        printf("Ungültige Angabe.");

    return 0;
}

```

Aufgabe 3: Programmieren in C

Sie spielen ein Würfelspiel mit 5 Würfeln. Schreiben Sie ein C-Programm, das aus 5 eingelesenen Augenzahlen ermittelt, ob eine der folgenden Situationen gewürfelt wurde und wenn ja, welche.

Grand	gleiche Augenzahl auf allen 5 Würfeln
Poker	gleiche Augenzahl auf 4 Würfeln
Full House	3 gleiche und 2 gleiche Augenzahlen
Nix gewonnen	Kein Grand, Poker oder Full House

Hinweis: Benutzen Sie für das Abspeichern der Augenzahlen ein Array.

Programm Würfelspiel

Würfel 1: ____

Würfel 2: ____

.

.

.

Würfel 5: ____

Die Ausgabe sieht dann so aus:

Es wurde gewürfelt:

Würfel	1	2	3	4	5
Augenzahl	x	x	x	x	x

Gewonnen! Grand (als Beispiel) bzw.

Nix gewonnen!

Lösung:

<Kopieren Sie bitte Ihr Programm hier hin. >

```

#include<stdio.h>
int main(){
    int i,j,summe=1,a[5];
    printf("Programm Würfelspiel\n\n");

    // Eingabe der Augenzahlen
    for(i=0;i<5;i++){
        printf("würfel %i:",i+1);
        scanf("%i",&a[i]);
    }

    // Ausgabe der Augenzahlen
    printf("\nEs wurde gewürfelt:\n\n");
    printf("würfel\t1\t2\t3\t4\t5\nAugenzahl");
    for(i=0;i<5;i++)
        printf("\t%i",a[i]);

    //Auswertung der Augenzahlen
    for(i=0; i<5; i++)
        for(j=0; j<5; j++)
            if(a[i]==a[j])
                summe++;

    //Ergebnisse des Programms
    printf("\n\n");
    if(summe==26)
        printf("Gewonnen! Grand");
    else if(summe==18)
        printf("Gewonnen! Poker\n");
    else if(summe==14)
        printf("Gewonnen! Full House \n");
    else if(summe<14)
        printf("Nichts gewonnen\n");

    return 0;
}

```