

Version A

Prüfung 02, Modul 319 "Applikationen entwerfen und implementieren"

Name: Vorname: Klasse: INF22.....

Punkte-Total: 50 Erreichte Punktzahl: Erreichte Note: $N=P/50*5+1$

Hilfsmittel: ohne Unterlagen Zeit: 60 Minuten

Aufgabe 01

[10 Punkte]

(Programmcode mit Verweigungen und Iterationen interpretieren und entwickeln können)

Betrachten Sie das untenstehende Programm und lösen sie die nachfolgenden Fragen

```
static void Main(string[] args) {  
    const int MAX = 50;  
    const int MIN = 10;  
    int i = MIN;  
    while (i <= MAX-MIN) {  
        Console.Write(i + " ");  
        if (i % 10 == 0)  
            Console.WriteLine(); // \t --> einen Zeilenumbruch  
        if (i >= 30)  
            break;  
        i += 5;  
    }  
    Console.ReadLine();  
}
```

Aufgabe 01a)

Was gibt das Programm auf dem Bildschirm aus?

10 ; (1)
15 ; 20 ; (2)
25 ; 30 ; (2)

5 Punkte

Aufgabe 01b)

Schreiben Sie das identische Programm, aber mit einer for-Schleife.

```
static void Main(string[] args) {  
    const int MAX = 50;  
    const int MIN = 10;  
  
    for (int i = MIN; i <= MAX-MIN ; i += 5) {  
        Console.Write(i + " ");  
        if (i % 10 == 0)  
            Console.WriteLine(); // \t --> einen Zeilenumbruch  
        if (i >= 30)  
            break;  
    }  
    Console.ReadLine();  
}
```

5 Punkte

Aufgabe 02

[11 Punkte]

(Programmcode mit Verweigungen und Iterationen interpretieren können)

Betrachten Sie das untenstehende Programm und lösen sie die nachfolgenden Fragen

```
static void Main(string[] args) {  
    int count =1;  
    double grade = 0.0, total =0.0, max=6.0;  
    while (count <=3) {  
        do{  
            Console.Write("Grade: ");  
        }while(Double.TryParse(Console.ReadLine(), out grade)==false);  
        total += grade;  
        if (grade < max)  
            max = grade;  
        count++;  
    }  
    Console.WriteLine("average grade" + total / count) );  
    Console.WriteLine("max. grade" + max);  
    Console.ReadLine();  
}
```

Aufgabe 01a)

Was gibt das Programm auf dem Bildschirm aus, wenn der Brutzer folgende Werte eingibt?

5 Punkte

Grade: 3.0

Grade: 5.0

Grade: 4.0

average grade : 3.0 (3)
max. grade : 3.0 (2)

Aufgabe 01b)

Das obenstehende Programm macht nicht das was es sollte. Können sie die zwei Fehler so beheben, dass auf die drei Eingaben 4.5, 5.5 und 5.0 die folgenden Ausgaben erscheinen:

6 Punkte

average grade : 5.0

max. grade : 5.5

Korrigieren Sie direkt:

```
int count =1; //count = 0;  
double grade = 0.0, total =0.0, max=6.0; // max=1.0;  
while (count <=3) { // while (count <3) {  
    do{  
        Console.Write("Grade: ");  
    }while(Double.TryParse(Console.ReadLine(), out grade)==false);  
    total += grade;  
    if (grade < max) // grade > max  
        max = grade;  
    count++;  
}  
Console.WriteLine("average grade" + total / (count-1) ); // Variante 2  
Console.WriteLine("max. grade" + max);
```

Aufgabe 03

[14 Punkte]

(Programmcode mit Verweigungen und Iterationen interpretieren und entwickeln können)

Betrachten Sie das untenstehende Programm und lösen sie die nachfolgenden Fragen

```
static void Main(string[] args) {
    const int DELTA = 20;
    Console.WriteLine("please enter a number: ");
    int no = Convert.ToInt32(Console.ReadLine());    no= 10
    for (int i = no + DELTA; i >= no; i=i-1) {        30 bis und mit 10 durchgehen und stets 1 abziehen
        Console.WriteLine(i + ", ");                30, 25, 20, 15, 14, 13, 12, 11, 10
        if (i > no+5) i>15
            i = i - 4;
    }
    Console.ReadLine();
}
```

Aufgabe 03a)

4 Punkte

Was wird ausgegeben, wenn der Benutzer den Wert 30 eingibt?

50, 45, 40, 35, 34, 33, 32, 31, 30 (4)

Aufgabe 03b)

4 Punkte

Was wird ausgegeben, wenn der Benutzer den Wert 40 eingibt?

60, 55, 50, 45, 44, 43, 42, 41, 40 (4)

Schreiben Sie das Programm so um, dass es statt der For-Schleife eine do-while Schleife enthält. Das Programm soll aber die selben Ausgaben wie oben liefern.

6 Punkte

```
static void Main(string[] args) {
    const int DELTA = 20;
    Console.WriteLine("please enter a number: ");
    int no = Convert.ToInt32(Console.ReadLine());

    int i = no + DELTA;    (2)
    do {
        Console.WriteLine(i + ", ");
        if (i > no+5)
            i = i - 4;
        i--;    (2)
    } while( i > no );    (2)
    Console.ReadLine();
}
```

Aufgabe 04

[15 Punkte]

(Programmcode mit Verweigungen und Iterationen interpretieren und entwickeln können)

Betrachten Sie die das untenstehende Programm. Es enthält fünf Verzweigungen.

```
static void Main(string[] args) {  
    int number1 =0, number2 =0;  
    Console.Write("please enter number 1: ");  
    number1 = Convert.ToInt32(Console.ReadLine());  
    Console.Write("please enter number 2: ");  
    number2 = Convert.ToInt32(Console.ReadLine());  
  
    //Verzweigung 1  
    if (number2 == number2 && number1 == number1)  
        Console.WriteLine("1");  
    //Verzweigung 2  
    if (number2 != number1 && number1 == number2)  
        Console.WriteLine("2");  
    //Verzweigung 3  
    if (number1 < 2* number2 && number1 > number2/2 )  
        Console.WriteLine("3");  
    //Verzweigung 4  
    if (number1 > number2 && number2 > number1)  
        Console.WriteLine("4");  
    //Verzweigung 5  
    if (number2 >= number1/2 || 2* number2 < number1)  
        Console.WriteLine("5");  
  
    Console.ReadLine();  
}
```

Aufgabe 04a)

Was gibt das Programm aus, wenn der Benutzer folgende Werte eingibt:

number 1 = 20
number 2 = 35

Ausgabe: 1 (5)
 3
 5

5 Punkte

Aufgabe 04b)

Bewerten Sie die Verzweigungen 1-5 nach folgenden Typen:

- a) OK (kann je nach Benutzereingabe true oder false sein)
- b) Immer true (egal was der Benutzer eingibt, die Bedingung gibt immer true)
- c) Immer false (egal was der Benutzer eingibt, die Bedingung gibt immer false)

Verzweigung	Typ (a, b, oder c) ?
Verweigung 1	b (2)
Verweigung 2	c (2)
Verweigung 3	a (2)
Verweigung 4	c (2)
Verweigung 5	b (2)

10Punkte

Anhang zur Prüfung (dürfen Sie als Zusammenfassung verwenden):

```
// Ausgabe von 1 bis 10 mit kopfgeprüfter Schleife
static void Main(string[] args) {
    int counter = 1;
    while (counter <= 10) {
        Console.WriteLine( counter);
        counter++;           //counter = counter +1
    }
    Console.ReadLine();
}
```

```
// Ausgabe von 1 bis 10 mit fussgeprüfter Schleife
static void Main(string[] args) {
    int counter = 1;
    do {
        Console.WriteLine(counter);
        counter++;
    } while (counter <= 10);
    Console.ReadLine();
}
```

```
// Ausgabe von 1 bis 10 mit Zählschleife
static void Main(string[] args) {
    for (int counter = 1; counter <= 10; counter++) {
        Console.WriteLine(counter);
    }
    Console.ReadLine();
}
```

```
// Anwendung einer switch-case Mehrfachverzweigung
static void Main(string[] args)
{
    string name = "";
    Console.Write("Geben Sie einen Monatsnamen ein: ");
    name = Console.ReadLine();
    switch (name)
    {
        case "Januar":
        case "Februar":
        case "März":
            Console.WriteLine("1. Quartal");
            break;
        case "April":
        ...
        default:
            Console.WriteLine("ungültiger Monatsname");
            break;
    }
}
```