

Übungsaufgaben A12 bis A14

Für die folgenden Aufgaben werden einige Operatoren in C benötigt. Diese müssen wie folgt im Programm notiert werden:

	In C:
Relation Gleichheit, =	==
Relation Ungleichheit, ≠	!=
Relationen <, ≤	<, <=
Relationen >, ≥	>, >=
Logisches UND	&&
Logisches ODER	
Negation NICHT(x)	!x
Modulo-Operation	%

A12

Es ist ein aussagenlogischer Ausdruck aufzustellen und in einem C-Programm zu benutzen. Es geht um eine Ermäßigung eines Eintrittspreises.

Ermäßigung für einen Eintrittspreis wird gewährt

- für Besucher die jünger als 16 Jahre sind.
(Variable *alter* mit einem ganzzahligen positiven Wert).
- für Rentner (Variable *r*, die mit *wahr* oder *falsch* belegt sein kann).
- für Studenten (Variable *s*, mit *wahr* oder *falsch*).

Wenn keine *Ermäßigung* gewährt wird, handelt es sich um einen *Vollzahler*.

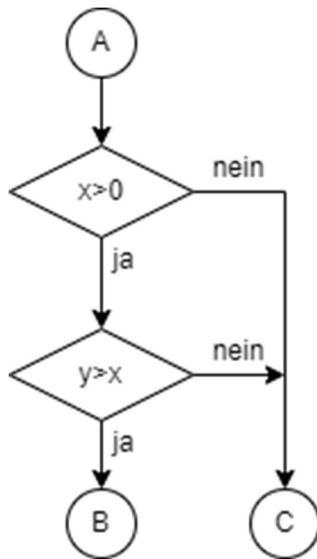
Aufgaben:

- a) Gesucht sind aussagenlogische Ausdrücke für *Ermäßigung* und für *Vollzahler*!
- b) Schreiben Sie die Ausdrücke aus a) mit den in der Programmiersprache gültigen Operatoren!
- c) Nutzen Sie die Ausdrücke aus b) in einem C-Programm und lassen Sie das Programm ausgehend von verschiedenen Variablenbelegungen über Ermäßigung bzw. Vollzahler entscheiden! Es sind die Werte für *alter*, *s* und *r* auszugeben, sowie die Information, ob Vollzahler oder Ermäßigung gilt.

Dabei sind für *alter* die ganzzahligen Werte 10, 13, 16, 19, 22, 25, 65 und 75 durchzuspielen. Die Variable *r* ist ausgehend von *alter* ab 67 auf den Logikwert *wahr* zu setzen und ist sonst *falsch*. Die Variable *s* ist für Werte von *alter* zwischen einschließlich 21 und 25 auf den Logikwert *wahr* zu setzen, und sonst auf *falsch*.

A13

Anordnungen von Selektionen im Steuerfluss können auch durch aussagenlogische Ausdrücke beschrieben werden. Im Folgenden werden Ihnen Ausschnitte von Programmablaufplänen für \wedge (UND), sowie für \vee (ODER) gegeben. Die einzelnen Aussagen entstehen dabei durch Relationen zwischen numerischen Variablen und Konstanten.



aussagenlogische Ausdrücke:

nach B wenn: $x > 0 \wedge y > x$

nach C wenn: $\neg(x > 0 \wedge y > x)$

C-Code gemäß PAP:

```

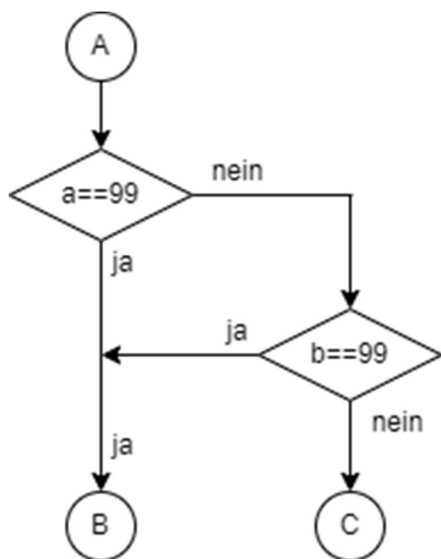
if (x > 0)
{
    if (y > x)
        printf("Konnektor B erreicht\n");
    else
        printf("Konnektor C erreicht\n");
}
else
    printf("Konnektor C erreicht\n");
  
```

C-Code mit aussagenlogischem

Bedingungs Ausdruck:

```

if (x > 0 && y > x)
    printf("Konnektor B erreicht\n");
else
    printf("Konnektor C erreicht\n");
  
```



aussagenlogische Ausdrücke:

nach B wenn: $a == 99 \vee b == 99$

nach C wenn: $\neg(a == 99 \vee b == 99)$

C-Code gemäß PAP:

```

if (a == 99)
    printf("Konnektor B erreicht\n");
else
{
    if (b == 99)
        printf("Konnektor B erreicht\n");
    else
        printf("Konnektor C erreicht\n");
}
  
```

C-Code mit aussagenlogischem Ausdruck:

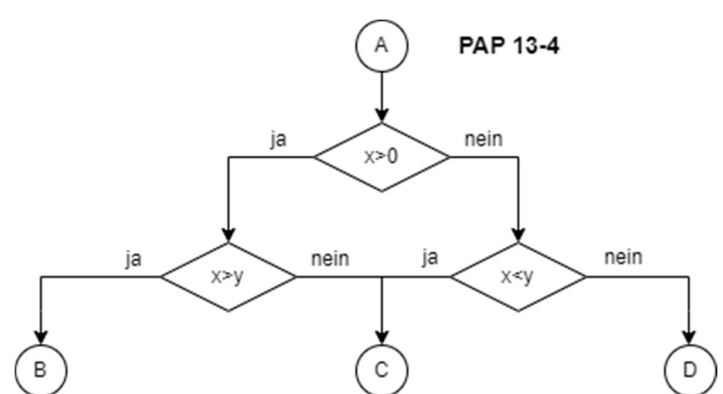
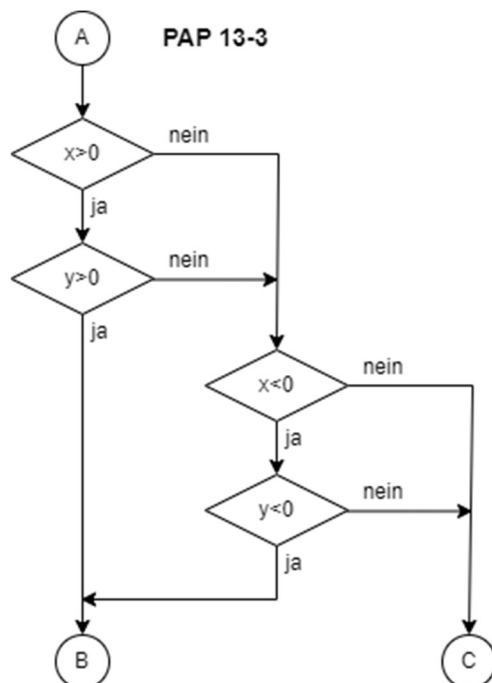
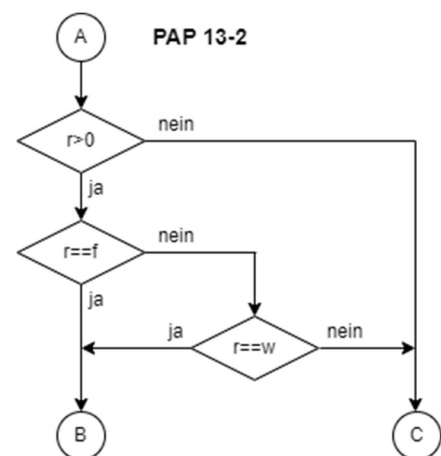
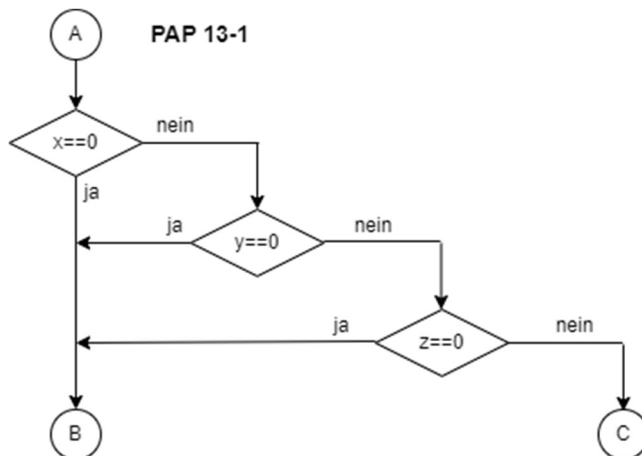
```

if (a == 99 || b == 99)
    printf("Konnektor B erreicht\n");
else
    printf("Konnektor C erreicht\n");
  
```

A13 Fortsetzung:

Geben Sie jeweils für die unten dargestellten PAP-Ausschnitte

- die aussagenlogischen Ausdrücke an, die dem Erreichen der unteren Konnektoren entsprechen.
- C-Quelltextausschnitte (keine vollständigen Programme) an. Bei Erreichen der Konnektoren ist eine entsprechende Ausgabe per printf() auszulösen



A14

Teilaufgaben:

- a) Schreiben Sie ein Programm (zuerst als PAP, dann als C-Programm), das eine Jahreszahl einliest und entscheidet, ob es sich um ein Schaltjahr handelt oder nicht! Die Regeln für die Schaltjahresentscheidung können im Web recherchiert werden, beispielsweise mit den Suchbegriffen: Schaltjahr, gregorianischer Kalender.
- b) Erweitern Sie Ihr Programm, dass zwei Jahreszahlen für Beginn und Ende einer Zeitspanne eingegeben werden können und die Schaltjahre in dieser Zeitspanne ausgegeben werden! Neben den Schaltjahren, ist die Anzahl der Schaltjahre in der Zeitspanne zu ermitteln und auszugeben.

Bei entsprechendem Wissenstand aus der Vorlesung bietet sich für die Schaltjahresentscheidung eine Funktion an.