

Es sind die vorgegebenen Datentypen zu beachten (int Werte werden in Variable vom Typ int eingelesen und double Werte in Variable vom Typ double).

Keine Zeichenketten oder Strings verwenden, wenn Zahlen verlangt sind.

Wenn ein Ergebnis berechnet werden soll, dann reicht die Ausgabe alleine (außer wenn nur diese explizit verlangt ist) nicht aus. Der Wert muss in einer Variable zur Verfügung stehen, sodass er prinzipiell für weitere Berechnungen verwendbar wäre.

Verwendung von globalen Variablen und goto ist (soweit sie nicht in der Angabe explizit empfohlen wird) untersagt.

Achten Sie darauf, dass die Objekte Ihrer Klasse in jedem Fall in einem konsistenten Zustand sein müssen.

Erweitern Sie die Klasse Auto (Evaluierungsaufgabe der Vorwoche) mit Operatoren, sodass folgende Operationen möglich sind:

```
Auto a, b, c;
bool slower = a < c; //true, falls a die
                    //kleinere Maximalge-
                    //schwindigkeit hat
a = 5 + b; //Ergibt ein neues Objekt vom
          //Typ Auto, dessen aktuelle
          //Geschwindigkeit um 5km/h
          //größer ist, als jene von b
```

Beachten Sie, dass der int Operand auch negative Werte annehmen kann und die Konsistenz Ihrer Klassenobjekte gewährleistet bleiben muss.

Die Angabe der Vorwoche:

Schreiben Sie eine Klasse Auto. Der Zustand eines Autos wird durch aktuelle und maximale Geschwindigkeit, Anzahl der Gänge und aktuell gewählter Gang beschrieben.

Schreiben Sie notwendige Konstruktoren, sowie Methoden, um einen bestimmten Gang auswählen zu können, um jeweils einen Gang hoch-, bzw. zurückzuschalten, um die Geschwindigkeit zu erhöhen („gasgeben“) und zu vermindern („bremsen“) und den aktuellen Zustand auf dem Bildschirm auszugeben.

Testen Sie die Funktionalität Ihrer Klasse mit einem entsprechenden Hauptprogramm.

Zusatzaufgabe (etwa 15 bis 30 Minuten extra): Schreiben Sie eine globale Funktion, die alle notwendigen Werte einliest, um ein neues Auto-Objekt zu erstellen und dieses neue Objekt an die aufrufende Funktion retourniert.

Eine mögliche Lösung:

```
#include <iostream>
#include <cstdlib>
using namespace std;

class Auto {
    int maxv;
    int v;
    int maxgear;
    int gear;
public:
    explicit Auto(int maxgear=4, int gear=0, int maxv=220, int v=0): maxv{abs(maxv)},
        v{v<0?0:v>abs(maxv)?abs(maxv):v}, maxgear{abs(maxgear)},
        gear{gear<0?0:gear>abs(maxgear)?abs(maxgear):gear} {
    }

    void shift(bool up) {
        if (up) {
            if (gear<maxgear) ++gear;
        }
        else
            if (gear>0) --gear;
    }

    void accel(int amount) {
        int res=v+amount;
        if (res<0) v=0;
        else if (res>maxv) v=maxv;
        else v=res;
    }

    void print() const {
        cout<<v<<"("<<maxv<<")km/h im "<<gear<<"("<<maxgear<<").Gang";
    }

};
```