

## Beispiel 1: Klasse Person (24 Punkte)

Ergänzen Sie die aus Übung 8 bekannte Klasse um die hier zusätzlich angegebenen Methoden, betten Sie Ihre Klasse in einen entsprechenden Namespace ein und erweitern Sie die Klasse für allem um ein detailliertes Exception Handling.

```
class Person {
    private: char *name;
            int age;

    public:  Person(void);
            Person(char*, int);
            Person(const Person&);
            ~Person();
            void Print(void);
            const Person &operator=(const Person&);
            const Person &operator=(char*);
            const Person &operator=(int);
            char &operator[](unsigned int);
            friend ostream &operator<<(ostream&, const Person&);
            friend istream &operator>>(istream&, Person&);
}

```

### Wichtig:

- Trennen Sie die Deklarationen und Definitionen voneinander, und speichern Sie diese in zwei Dateien namens person.h und person.cpp.
- Implementieren Sie Ihre Klasse innerhalb eines Namespace. Die Namesgebung ist für Sie frei wählbar.
- Alle Ausnahmesituationen (Bereichsüberschreitungen, Allokationsfehler, etc), die auftreten könnten sind mittels exception handling (try, throw, catch) abzufangen!!!  
 Z.B. try {.....if(!string) throw(Allokationsfehler()); } catch(Allokationsfehler) {.....}

### Standards:

Es gelten die üblichen Standards.

**Abgabe:** bis 13.6. 23:59 Uhr elektronisch (Verzeichnis uebung9 einrichten!) und am 14.6. zu Beginn der Klausur auf Papier (Listing der Quelldateien).