

Klausurvorbereitung

*Zeiger allgemein,
Was bedeutet: &variable, *zeiger etc.?*

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter& am) {  
    am.geh = 44;  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter m1(4);  
    help( ??? m1);  
}
```

1. help(m1);
2. help(&m1);
3. help(*m1);
4. help(** m1);

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter& am) {  
    am.geh = 44;  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter m1(4);  
    help( ??? m1);  
}
```

1. help(m1);
2. help(&m1);
3. help(*m1);
4. help(** m1);

Ergebnis:

 1 2 3 4

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter am) {  
    am.geh = 44;  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter m1(4);  
    help( ??? m1);  
}
```

1. help(m1);
2. help(&m1);
3. help(*m1);
4. help(** m1);

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter am) {  
    am.geh = 44;  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter m1(4);  
    help( ??? m1);  
}
```

1. help(m1);
2. help(&m1);
3. help(*m1);
4. help(** m1);

Ergebnis:

 1 2 3 4

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter& am) {  
    am.geh = 44;  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter m1(4);  
    help( ??? m1);  
}
```

1. help(m1);
2. help(&m1);
3. help(*m1);
4. help(** m1);

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter& am) {  
    am.geh = 44;  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter m1(4);  
    help( ??? m1);  
}
```

1. help(m1);
2. help(&m1);
3. help(*m1);
4. help(** m1);

Ergebnis:

1 2 3 4

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter am) {  
    am.geh = 44;  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter* p = new Mitarbeiter(4);  
    help( ???p);  
}
```

1. help(m1) ;
2. help(&m1);
3. help(*m1);
4. help(** m1);

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter am) {  
    am.geh = 44;  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter* p = new Mitarbeiter(4);  
    help( ???p);  
}
```

1. help(m1);
2. help(&m1);
3. help(*m1);
4. help(** m1);

Ergebnis:

__ 1 __ 2 _ _ 3 __ 4

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter* pm) {  
    pm->geh = 44;  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter* p = new Mitarbeiter(4);  
    help( ???p);  
}
```

1. help(m1) ;
2. help(&m1);
3. help(*m1);
4. help(** m1);

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter* pm) {  
    pm->geh = 44;  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter* p = new Mitarbeiter(4);  
    help( ???p);  
}
```

1. help(m1);
2. help(&m1);
3. help(*m1);
4. help(** m1);

Ergebnis:

1 2 3 4

Folie korrigiert, 20.12.2021

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter& am) {  
    am.geh = 44;  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter* p = new Mitarbeiter(4);  
    help( ???p);  
}
```

1. help(m1) ;
2. help(&m1);
3. help(*m1);
4. help(** m1);

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter& am) {  
    am.geh = 44;  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter* p = new Mitarbeiter(4);  
    help( ???p);  
}
```

1. help(m1);
2. help(&m1);
3. help(*m1);
4. help(** m1);

Ergebnis:

__ 1 __ 2 _ _ 3 __ 4

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter am) {  
    am.geh = 44;  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter m1(4);  
    help( ??? m1);  
}
```

1. help(m1);
2. help(&m1);
3. help(*m1);
4. help(** m1);

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter am) {  
    am.geh = 44;  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter m1(4);  
    help( ??? m1);  
}
```

1. help(m1);
2. help(&m1);
3. help(*m1);
4. help(** m1);

Ergebnis:

 1 2 3 4

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter* am) {  
    am = new Mitarbeiter(300);  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter m1(4);  
    help( ??? m1);  
}
```

1. help(m1);
2. help(&m1);
3. help(*m1);
4. help(** m1);

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout <<"-M " << geh;}  
};
```

```
void help(Mitarbeiter* am) {  
    am = new Mitarbeiter(300);  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter m1(4);  
    help( ??? m1);  
}
```

1. help(m1);
2. help(&m1);
3. help(*m1);
4. help(** m1);

Ergebnis:

__ 1 _ _ 2 __ 3 __ 4

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout <<"-M " << geh;}  
};
```

```
void help(Mitarbeiter*& am) {  
    am = new Mitarbeiter(300);  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter m1(4);  
    Mitarbeiter* p = &m1;  
    help( ????? p);  
    cout << p->geh << " ";  
}
```

1. help(p) ;
2. help(&p);
3. help(*p);
4. help(*& p);

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter*& am) {  
    am = new Mitarbeiter(300);  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter m1(4);  
    Mitarbeiter* p = &m1;  
    help(????? p);  
    cout << p->geh << " ";  
}
```

1. help(p) ;
2. help(&p);
3. help(*p);
4. help(*& p);

Ergebnis:

1 2 3 4

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter&& am) {  
    ...  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter m1(4);  
    Mitarbeiter* p = &m1;  
    help( ????? p);  
    cout << p->geh << " ";  
}
```

1. help(p) ;
2. help(move(&p));
3. help(move(p));
4. help(move(*p));

Ergebnis:

__ 1 __ 2 __ 3 _ _ 4

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
void help(Mitarbeiter&& am) {  
    ...  
    cout << "Ende";  
}  
void f1() {  
    Mitarbeiter m1(4);  
    Mitarbeiter* p = &m1;  
    help( Mitarbeiter(6));  
    cout << p->geh << " ";  
}
```

1. help(p) ;
2. help(move(&p));
3. help(move(p));
4. help(move(*p));

Ergebnis:

__ 1 __ 2 __ 3 __ 4

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout << "-M " << geh;}  
};
```

```
typedef Mitarbeiter* XXX;  
void help(XXX& am) {  
    am = new Mitarbeiter(300);  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter m1(4);  
    XXX p = &m1;  
    help( ????? p);  
    cout << p->geh << " ";  
}
```

1. help(p) ;
2. help(&p);
3. help(*p);
4. help(*& p);

Clicker: Was wird für „?“ eingesetzt.?

```
class Mitarbeiter {  
public:  
    int geh;  
    Mitarbeiter(int g) {geh = g;  
        cout << "+M " << geh;}  
    ~Mitarbeiter() {  
        cout <<"-M " << geh;}  
};
```

```
typedef Mitarbeiter* XXX;  
void help(XXX& am) {  
    am = new Mitarbeiter(300);  
    cout << "Ende";  
}  
void f() {  
    Mitarbeiter m1(4);  
    XXX p = &m1;  
    help( ????? p);  
    cout << p->geh << " ";  
}
```

1. help(p) ;
2. help(&p);
3. help(*p);
4. help(*& p);

Ergebnis:

1 2 3 4