

Lösung:

- a)

```
void Umdrehen() {  
    LinksDrehen();  
    LinksDrehen();  
}
```
- b)

```
void SideStepRechts() { //rechts entsprechend  
    RechtsDrehen();  
    Schritt();  
    LinksDrehen();  
}
```
- c)

```
void SchrittZurück() {  
    Umdrehen();  
    Schritt();  
    Umdrehen();  
}
```
- d)

```
void Schwindelig() {  
    for (int i=1;i<80;i++){  
        LinksDrehen();  
    }  
}
```
- e)

```
void Drehzahl(int anz){  
    for (int i=1;i<anz;i++){  
        LinksDrehen();  
    }  
}
```
- f)

```
void ViererReiheLegen() {  
    for (int i=0;i<4;i++){  
        Hinlegen();  
        Schritt();  
    }  
}
```
- g)

```
void ZurWandGehen() {  
    while (!IstWand()){  
        Schritt();  
    }  
}
```

```

h) void ReiheZurWandLegen(){
    while (!IstWand()){
        Hinlegen();
        Schritt();
    }
}

i) void FuenferQuadrat(){
    for (int i=0;i<4;i++){
        ViererReiheLegen();
        LinksDrehen();
    }
}

j) void MauerDeluxe(int breite, int hoehe){
    for (int i=1; i<=breite; i++){
        for (int k=1; k<=hoehe; k++){
            Hinlegen();
        }
        SideStepLinks();
    }
}

k) void Invertieren(){
    if (IstZiegel()){
        Aufheben()
    }else{
        Hinlegen();
    }
}

l) void InvertiereBisWand(){
    while (!IstWand()){
        Invertieren();
        Schritt();
    }
    Schritt();
}

```

```

m) void ZiegelRand(int anz){
    for (int i = 1; i <= anz; i++){
        if (IstWand()){
            LinksDrehen();
        }
        Hinlegen();
        Schritt();
    }
}

n) void InDieEcke(){
    while (!IstBlickNorden()){
        RechtsDrehen();
    }
    while (!IstWand()){
        Schritt();
    }
    LinksDrehen();
    while (!IstWand()){
        Schritt();
    }
    LinksDrehen();
}

```