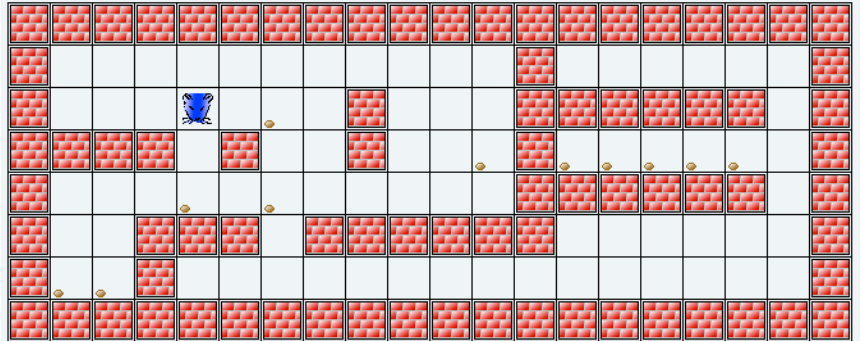


Anweisungen und Programme

Beispiel Band 1 Auflage 3 und 4 Kapitel 7 Beispielprogramm 1 Hamsterprogramm 1
Aufgabe: Der Hamster soll zwei Körner in dem vorgegebenen Territorium fressen

```
void main() {  
  
    // friss erstes Korn  
    vor();  
    vor();  
    nimm();  
  
    // friss zweites Korn  
    linksUm();  
    vor();  
    vor();  
    nimm();  
}
```



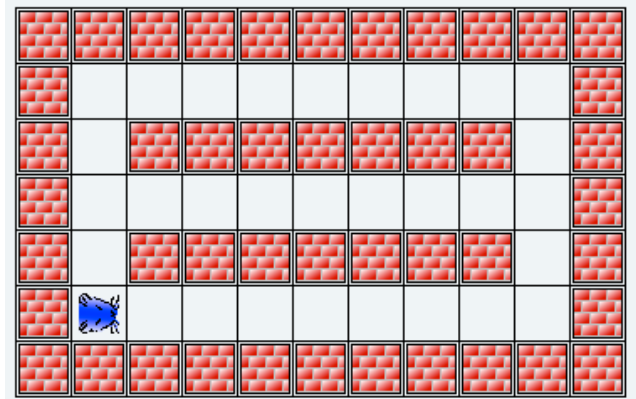
Übung:

Ändere das Programm so ab, dass der Hamster vier statt zwei Körner frisst

Prozeduren

Beispiel Band 1 Auflage 3 und 4 Kapitel 8 Beispielprogramm 2 Hamsterprogramm
Aufgabe: Der Hamster soll auf allen Feldern in dem vorgegebenen Territorium jeweils genau ein Korn ablegen. Der Hamster hat mindestens 31 Körner im Maul.

```
void main() {  
  
    // bearbeite die unterste Reihe  
    bearbeiteEineReihe();  
  
    // begib dich zur mittleren Reihe und bearbeite sie  
    linksUm();  
    vor();  
    gibUndVor();  
    linksUm();  
    bearbeiteEineReihe();  
  
    // bearbeite das Feld unter dir  
    linksUm();  
    vor();  
    gib();  
    kehrt();  
    vor();  
  
    // begib dich zur obersten Reihe und bearbeite sie  
    vor();  
    gibUndVor();  
    rechtsUm();  
}
```



```

    bearbeiteEineReihe();

    // bearbeite letztes Feld
    rechtsUm();
    vor();
    gib();
}

void bearbeiteEineReihe() {
    gibUndVor();
    gibUndVor();
    gibUndVor();
    gibUndVor();
    gibUndVor();
    gibUndVor();
    gibUndVor();
    gibUndVor();
    gib();
}

void gibUndVor() {
    gib();
    vor();
}

void kehrt() {
    linksUm();
    linksUm();
}

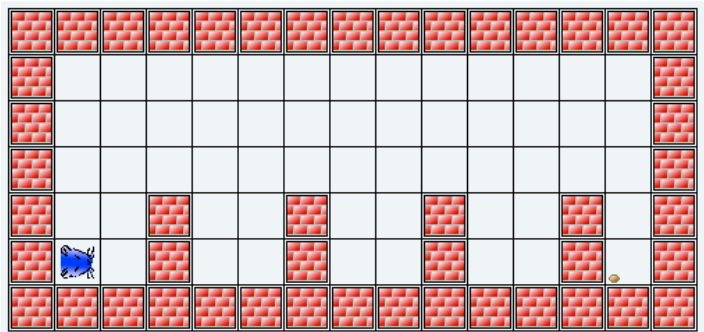
void rechtsUm() {
    kehrt();
    linksUm();
}

```

Übung:

Ändern Sie die Lösung der Aufgabe dahingehend, dass der Hamster auf jede zweite Kachel ein Korn ablegt.

Vom vielen Körner fressen ist der Hamster doch recht dick geworden und sein Arzt hat ihm geraten, Sport zu treiben. Diesen Ratschlag hat der Hamster befolgt und sich sogar für die Olympischen Spiele qualifizieren können, und zwar für den 110 Meter Hürdenlauf. Nun ist er fleißig am Trainieren. Zur Belohnung und zum Ansporn hat er sich am Ende des Parcours ein „Diät-Korn“ hingelegt. Lassen Sie den Hamster die Hürden überqueren und das Korn fressen.



























































Beispiel Band 1 Auflage 3und4 Kapitel9 Beispielprogramm3 Hamsterprogramm

```
void main() {
```

```
// obere Reihe
ueberpruefeEineReihe();
```

}

```
void evtlFressen() {
```

```

nimm();

/*
 * falls es das einzige Korn auf dem Feld war,
 * muss es wieder abgelegt werden
 */
if (!kornDa()) {
    gib();
}
}

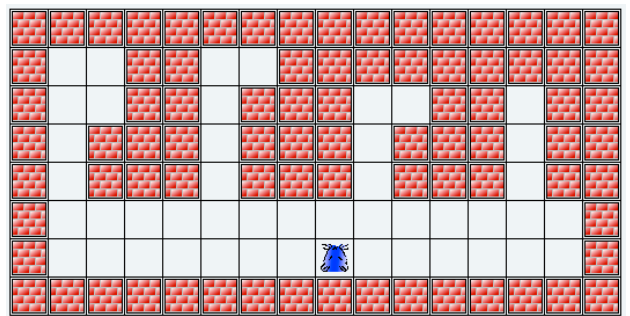
void rechtsUm() {
    linksUm();
    linksUm();
    linksUm();
}

```

Übung

Passen Sie die Lösung von dem Beispiel an folgende Ausgangssituation an: Auf jedem Feld liegen ein, zwei oder drei Körner.

Der Hamster steht in einem Territorium mit vier unterschiedlich großen Körnerkammern mit vier, drei, zwei bzw. einer leeren Kachel. Der Hamster kann anfangs ein bis vier Körner im Maul haben. Abhängig von dieser Zahl soll er auf jede Kachel der entsprechend großen Körnerkammer ein Korn ablegen.



Wiederholungsanweisungen

Beispiel Band 1 Auflage 3 und 4 Kapitel 10 Beispielprogramm 1 Hamsterprogramm
 Aufgabe: In einem rechteckigen geschlossenen Raum unbekannter Größe ohne innere Mauern sind wahllos eine unbekannte Anzahl an Körnern verstreut. Der Hamster, der sich zu Anfang in der linken unteren Ecke des Raumes mit Blickrichtung Ost befindet, soll alle Körner aufsammeln und dann stehen bleiben.

```

void main() {
    ernteEineReiheUndLaufeZurueck();
    rechtsUm();
    while (vornFrei()) {
        vor();
        rechtsUm();
        ernteEineReiheUndLaufeZurueck();
        rechtsUm();
    }
}

```

```

void ernteEineReiheUndLaufeZurueck() {
    ernteEineReihe();
    kehrt();
    laufeZurueck();
}

```

```

void ernteEineReihe() {
    sammle();
    while (vornFrei()) {
        vor();
        sammle();
    }
}

```

```

void laufeZurueck() {
    while (vornFrei()) {
        vor();
    }
}

```

```

void sammle() {
    while (kornDa()) {
        nimm();
    }
}

```

```

void rechtsUm() {
    kehrt();
    linksUm();
}

```

```

void kehrt() {
    linksUm();
    linksUm();
}

```

Übung:

Im Unterschied zum Beispielprogramm soll der Hamster keine Körner aufsammeln, sondern ablegen. Seine Aufgabe lautet: In einem rechteckigen geschlossenen Raum unbekannter Größe ohne innere Mauern sind keine Körnern verstreut. Der Hamster, der sich zu Anfang in der linken unteren Ecke des Raumes mit Blickrichtung Ost befindet, soll alle Kacheln des Raumes genau ein Korn ablegen und dann stehen bleiben. Der Hamster hat mindestens so viele Körner im Maul, wie es Kacheln im Raum gibt.

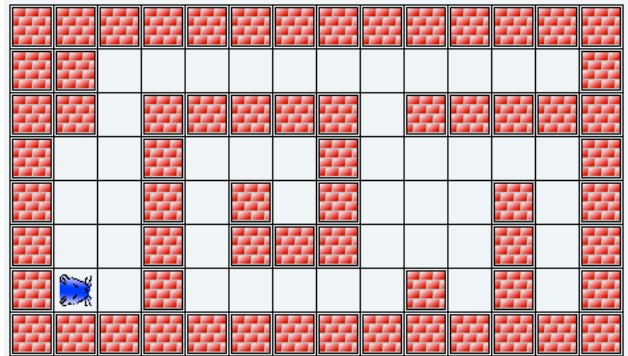
Der Hamster steht vor einer regelmäßigen Mulde unbekannter Tiefe. es liegen keine Körner im Territorium. Der Hamster, der anfangs genau ein Korn im Maul hat, soll bis zur tiefsten Stelle der Mulde hinabsteigen, sich umdrehen, wieder hinaufsteigen und an seiner Ausgangsposition stehen bleiben.

Boolesche Funktionen

Beispiel Band 1 Auflage 3und4 Kapitel11 Beispielprogramm3 Hamsterprogramm
Aufgabe: Der Hamster, der genau ein Korn im Maul hat, befindet sich in einem geschlossenen, körnerlosen Raum unbekannter Größe. Rechts von ihm befindet sich eine wand und vor ihm die Kachel ist frei. Der Hamster solll solange gegen den Uhrzeigersinn an der Wand entlanglaufen, bis er irgendwann wieder seine Ausgangskachel erreicht.

```
void main() {
    gib(); // markiere Ausgangsposition
    vor();

    /*
     * irgendwann kehrt der Hamster in seine
     * Ausgangsposition zurueck
     */
    while (!kornDa()) {
        if (rechtsFrei()) {
            rechtsUm();
            vor();
        } else if (vornFrei()) {
            vor();
        } else if (linksFrei()) {
            linksUm();
            vor();
        } else {
            kehrt();
            vor();
        }
    }
}
```



```
boolean rechtsFrei() {
    rechtsUm();
    if (vornFrei()) {
        linksUm();
        return true;
    } else {
        linksUm();
        return false;
    }
}
```

```
boolean linksFrei() {
    linksUm();
    if (vornFrei()) {
        rechtsUm();
        return true;
    } else {
        rechtsUm();
    }
}
```

```

        return false;
    }
}

void rechtsUm() {
    kehrt();
    linksUm();
}

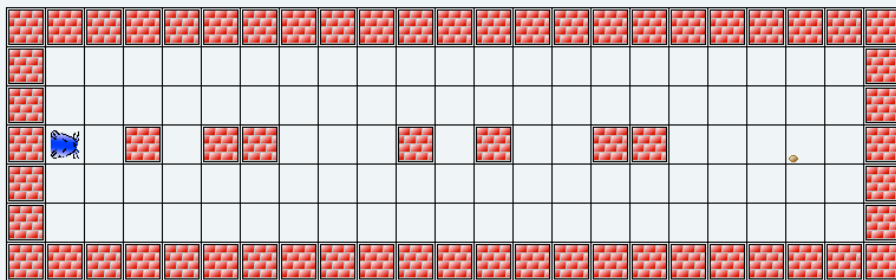
void kehrt() {
    linksUm();
    linksUm();
}

```

Übung

Erweitern Sie die Lösung von dem Beispiel derart, dass der Hamster die Strecke zunächst gegen den Uhrzeigersinn absolviert und sie danach noch einmal im Uhrzeigersinn zurückläuft.

Der Hamster ist zum Skifahrer mutiert. Vor ihm befindet sich ein Slalomparcours mit einem Korn am Ende, das das Ziel Markiert. Der Hamster soll den Slalomparcours bewältigen und das Korn fressen.



Boolesche Variablen

Beispiel Band 1 Auflage 3 und 4 Kapitel 13 Beispielprogramm 1 Hamsterprogramm
Aufgabe: Der Hamster steht irgendwo im Territorium. Er soll bis zur nächsten Wand laufen und dabei alle Körner, die er unterwegs findet, einsammeln. Nur wenn er eine gerade Anzahl an Körnern oder kein Korn gesammelt hat, soll er alle Körner, die er im Maul hat, an der Mauer ablegen. Ansonsten soll er nichts weiter tun.

Lösung 1 lokale Variable

```

void main() {
    boolean geradeAnzahl = true; // 0 Körner eingesammelt
    geradeAnzahl = sammle();
    while (vornFrei()) {
        vor();
        boolean gesammelt = sammle();

        /*
        * gerade Zahl + gerade Zahl = gerade Zahl

```

```

    * ungerade Zahl + ungerade Zahl = gerade Zahl
    * alle anderen Faelle ergeben eine ungerade Zahl
    */
    geradeAnzahl =
        (geradeAnzahl && gesammelt) ||
        (!geradeAnzahl && !gesammelt);
}

/*
 * falls gerade Anzahl an Koernern gefunden:
 * alle Koerner ablegen
 */
if (geradeAnzahl) {
    while (!maulLeer()) {
        gib();
    }
}

/*
 * sammle alle Koerner auf aktuellem Feld auf; falls gerade
 * Anzahl an Koernern eingesammelt wurde, liefere true;
 * ansonsten liefere false
 */
boolean sammle() {
    boolean geradeAnzahl = true; // 0 Koerner eingesammelt
    while (kornDa()) {
        nimm();

        /*
         * auf jede gerade folgt eine ungerade
         * Zahl und umgekehrt
         */
        geradeAnzahl = !geradeAnzahl;
    }
    return geradeAnzahl;
}

```

Lösung 2 globale Variable

// globale Variable

boolean geradeAnzahl = true; // 0 Koerner eingesammelt

```

void main() {
    sammle();
    while (vornFrei()) {
        vor();
        sammle();
    }
    if (geradeAnzahl) {
        while (!maulLeer()) {

```



```

        gib();
    }
}

// sammle alle Koerner auf aktuellem Feld auf
void sammle() {
    while (kornDa()) {
        nimm();

        /*
        * auf jede gerade folgt eine ungerade
        * Zahl und umgekehrt
        */
        geradeAnzahl = !geradeAnzahl;
    }
}

```

Übung:

Ändern Sie die Lösungen so ab, dass der Hamster nur, wenn er eine ungerade Anzahl an Körnern eingesammelt hat, alle Körner, die er im Maul hat, an der Mauer ablegt.

Der Hamster steht mit Blickrichtung Ost in der oberen linken Ecke eines körnerlosen durch Mauern abgeschlossenen rechteckigen Raumes, ohne innere Mauern. Er hat eine unbekannte Anzahl Körner im Maul. Seine Aufgabe besteht darin, dass er solange er noch Körner im Maul hat in seinem Territorium die Körner im Schachbrettmuster auslegt.

Zahlen, Variablen und Ausdrücke

Beispiel Band 1 Auflage 3 und 4 Kapitel 14 Beispielprogramm 3

Aufgabe: Der Hamster befindet sich in einem geschlossenen, rechteckigen Raum, ohne innere Mauern unbekannter Größe. Auf irgendeinem Feld des Territoriums liegt ein Korn. Der Hamster befindet sich anfangs in der linken unteren Ecke mit Blickrichtung Ost. Der Hamster bekommt die Aufgabe das Korn zu finden und zu fressen.

```
int radius = 1; // speichert die Groesse des aktuellen Radius
```

```

void main() {
    while (!kornDa()) {
        testeEinenKreis();

        // nach jeder Runde wird der Radius ein Feld groesser
        radius++;
    }
    nimm();
}

void testeEinenKreis() {
    int richtungen = 0;

```

```

// ein Kreis besteht aus vier Richtungen
while (!kornDa() && (richtungen < 4)) {
    testeEineRichtung();
    richtungen++;
}
}

void testeEineRichtung() {
    int schritte = 0;

    /*
    * die Ueberpruefung einer Richtung besteht aus der
    * Ueberpruefung von so vielen Feldern, wie der Radius
    * des Kreises aktuell gross ist; die zusaetzliche
    * Konjunktion mit vornFrei() ist notwendig, falls das
    * Feld nicht quadratisch ist
    */
    while (!kornDa() && (schritte < radius) && vornFrei()) {
        vor();
        schritte++;
    }

    if (!kornDa()) {
        linksUm();
    }
}

```

Übung

Erweitern Sie das Programm derart, dass der Hamster, sobald er das Korn gefunden hat, auf dem schnellsten Weg (d.h. mit möglichst wenig vor-Befehlen) in seine Ausgangsecke wider zurückkehrt.

Der Hamster steht vor einem Gebirge mit unregelmäßigen Bergen unbekannter Höhe. Er bekommt die Aufgabe, das Gebirge zu übersteigen. Sobald er das Gebirge überstiegen hat, d.h. sobald er auf eine Ebene gelangt, die dieselbe Höhe aufweist wie seine Ausgangsposition des Hamsters, soll er stehen bleiben. Der Hamster hat keine Körner im Maul und im Territorium befinden sich auch keine Körner.

Prozeduren und Funktionen

Beispiel Band 1 Auflage 3 und 4 Kapitel 15 Beispielprogramm 1

Aufgabe: Der Hamster steht direkt vor einem regelmäßigen Berg unbekannter Höhe. Er selbst hat keine Körner im Maul. Auf der Kachel, auf der er steht, liegt eine bestimmte Anzahl an Körnern. Ansonsten liegen keine Körner im Feld. Der Hamster soll den Berg erklimmen und dabei solange wie möglich auf jeder Stufe ein Korn ablegen. Er soll jedoch keinen unnötigen Ballast mitschleppen.

```

void main() {
    int stufenAnzahl = zaehleStufen();

```

```

// nimm genugend Koerner ins Maul
while ((stufenAnzahl > 0) && kornDa()) {
    nimm();
    stufenAnzahl--;
}

erklimmeBerg();
}

// ermittle die Hoehe des Berges (ohne Seiteneffekte)
int zaehleStufen() {

    // erklimme die einzelnen Stufen und vermerke die Anzahl
    int anzahl = 0;
    while (!gipfelErreicht()) {
        erklimmeStufe();
        anzahl++;
    }

    // und wieder hinunter (Vermeidung von Seiteneffekten)
    kehrt();
    int schritte = anzahl;
    while (schritte > 0) {
        klettereStufeHinunter();
        schritte--;
    }
    kehrt();
    return anzahl;
}

boolean gipfelErreicht() {
    return vornFrei();
}

void erklimmeStufe() {
    linksUm();
    vor();
    rechtsUm();
    vor();
}

void klettereStufeHinunter() {
    vor();
    linksUm();
    vor();
    rechtsUm();
}

void erklimmeBerg() {
    while (!gipfelErreicht()) {

```

```

    erklimmeStufe();
    if (!maulLeer()) {
        gib();
    }
}
}

```

```

void rechtsUm() {
    kehrt();
    linksUm();
}

```

```

void kehrt() {
    linksUm();
    linksUm();
}

```

Übung

Die Aufgabenstellung wird derart verändert, dass der Berg nicht unbedingt regelmäßig sein muss.

Der Hamster befindet sich in einem beliebig großen rechteckigen geschlossenen Territorium ohne innere Mauern und ohne Körner auf den Kacheln. Solange er Körner im Maul hat, soll er auf die äußeren Kacheln jeweils ein Korn ablegen, auf den zweitäußeren zwei auf den drittäußeren drei usw. .

Funktionsparameter

Beispiel Band 1 Auflage 3 und 4 Kapitel 16 Beispielprogramm 2

Aufgabe: Der Hamster befindet sich in einem geschlossenen, rechteckigen Raum, ohne innere Mauern unbekannter Größe. Auf irgendeinem Feld des Territoriums liegt ein Korn. Der Hamster befindet sich anfangs in der linken unteren Ecke mit Blickrichtung Ost. Der Hamster bekommt die Aufgabe das Korn zu finden und zu fressen.

```

void main() {

    /*
    * lokale Variable;
    * speichert die Groesse des aktuellen Radius
    */
    int radius = 1;
    while (!kornDa()) {

        // radius wird als Parameter uebergeben
        testeEinenKreis(radius);

        // nach jeder Runde wird der Radius ein Feld groesser
        radius++;
    }
    nimm();
}

```

```

void testeEinenKreis(int radius) {
    int richtungen = 0;

    // ein Kreis besteht aus vier Richtungen
    while (!kornDa() && (richtungen < 4)) {
        testeEineRichtung(radius);
        richtungen++;
    }
}

void testeEineRichtung(int radius) {
    int schritte = 0;

    /*
    * die Ueberpruefung einer Richtung besteht aus der
    * Ueberpruefung von so vielen Feldern, wie der Radius
    * des Kreises aktuell betraegt
    */
    while (!kornDa() && (schritte < radius) && vornFrei()) {
        vor();
        schritte++;
    }

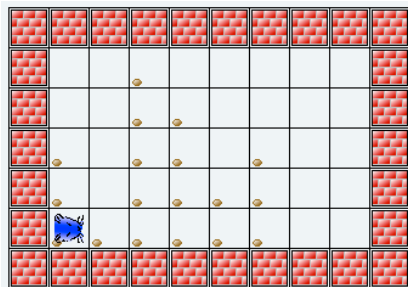
    if (!kornDa()) {
        linksUm();
    }
}

```

Übung

Der Hamster steht in der linken unteren Ecke (Blickrichtung Ost) eines durch Mauern abgeschlossenes mauerlosen rechteckigen Territoriums unbekannter Größe. Im Territorium befinden sich vertikale Körnertürme. Die erste leere Spalte markiert das Ende der Körnertürme. Der Hamster bekommt die Aufgabe, die Körnertürme so zu sortieren, dass sie zum Schluss in aufsteigender Größe im Territorium angeordnet sind.

Vorher:



Nachher:

