Master – Einfügen & Entfernen im Array (Deutsch + فارسی)

Dieses Dokument fasst typische Muster zum Einfügen und Entfernen von Elementen in Arrays zusammen: Einfügen an Position, Entfernen an Position, Einfügen/Entfernen am Ende, Varianten mit Objekten, Versionen mit Bedingungen sowie typische Fehler in IHK-Prüfungen (GA2).

این فایل الگوهای مهم برای درج (Einfügen) و حذف (Entfernen) عناصر در آرایه را جمع‌بندی می‌کند: درج در یک موقعیت خاص، حذف از یک موقعیت، درج/حذف در انتهای آرایه، نسخه‌های مخصوص آبجکت‌ها، نسخه‌های شرطی و خطاهای رایج در امتحان‌های IHK (GA2).

---

1. Einfügen an Position – درج در یک موقعیت مشخص

1.1 Integer einfügen an Position pos (mit rechts-Shift)

```
prozedur fuegeEin(a: Array von Integer, n: Integer, pos: Integer, wert: Integer)
  // a: Array mit Kapazität
  // n: aktuelle Anzahl der Elemente
  // pos: Einfügeposition (0 .. n)
  // wert: neuer Wert

  // Elemente nach rechts schieben
  für i von n - 1 bis pos schritt -1
      a[i + 1] = a[i]
  ende für

  // neuen Wert einfügen
  a[pos] = wert

  // Anzahl erhöhen (wenn n als Referenz übergeben wird)
endprozedur
```

در این الگو، ابتدا عناصر از انتها تا موقعیت **pos** یک خانه به سمت راست شیفت داده می‌شوند (**a[i + 1] = a[i]**)، سپس مقدار جدید در **a[pos]** قرار می‌گیرد. این اساس درج در آرایه است.

1.2 Artikel an Position pos einfügen

```
prozedur fuegeArtikelEin(artikel: Array von Artikel, n: Integer, pos:
                                                    Integer, neu: Artikel)
  für i von n - 1 bis pos schritt -1
      artikel[i + 1] = artikel[i]
  ende für
  artikel[pos] = neu
endprozedur
```

برای آبجکت‌ها (مثل **Artikel**) منطق دقیقاً همان است، فقط نوع داده فرق می‌کند.

2. Entfernen an Position – حذف از یک موقعیت مشخص

2.1 Integer an Position pos entfernen (mit links-Shift)

```
prozedur entferneAnPos(a: Array von Integer, n: Integer, pos: Integer)
  // a: Array
  // n: aktuelle Anzahl
  // pos: zu löschende Position (0 .. n - 1)

  für i von pos bis n - 2
      a[i] = a[i + 1]
  ende für

  // optional: letztes Element "leeren"
  // a[n - 1] = 0

  // n um 1 verringern (wenn n als Referenz übergeben wird)
endprozedur
```

2.2 Artikel an Position pos entfernen

```
prozedur entferneArtikel(artikel: Array von Artikel, n: Integer, pos: Integer)
  für i von pos bis n - 2
      artikel[i] = artikel[i + 1]
  ende für
  // optional: artikel[n - 1] = null
endprozedur
```

در حذف، برعکس درج، از موقعیت **pos** تا قبل از آخرین عنصر، هر عنصر با عنصر بعدی‌اش جایگزین می‌شود (**a[i] = a[i + 1]**). در پایان، عنصر آخر دیگر معتبر نیست.

3. Einfügen & Entfernen am Ende – درج / حذف در انتهای آرایه

3.1 Einfügen am Ende (Append)

```
  prozedur fuegeHintenAn(a: Array von Integer, n: Integer, wert: Integer)
    a[n] = wert
    // n = n + 1 (wenn n als Referenz übergeben wird)
endprozedur
```

3.2 Entfernen am Ende (Pop)

```
  prozedur entferneHinten(a: Array von Integer, n: Integer)
    if n = 0 then
        return
    end if
    // optional: a[n - 1] = 0
    // n = n - 1
endprozedur
```

درج در انتهای آرایه ساده‌ترین حالت است: مقدار جدید در `a[n]` قرار می‌گیرد و تعداد عناصر یک واحد زیاد می‌شود. حذف از انتها فقط با کم کردن `n` انجام می‌شود.

4. Einfügen / Entfernen mit Bedingung – درج / حذف شرطی

4.1 Einfügen nur, wenn Platz frei ist

```
prozedur fuegeEinWennPlatz(a: Array von Integer, n: Integer, maxGroesse:
                                        Integer, wert: Integer)
  if n >= maxGroesse then
      // kein Platz mehr
      return
  end if
  // am Ende einfügen
  a[n] = wert
  // n = n + 1
endprozedur
```

4.2 Entfernen nur, wenn Wert gefunden wird

```
prozedur entferneErstesVorkommen(a: Array von Integer, n: Integer, wert: Integer)
  index = -1
  für i von 0 bis n - 1
      if a[i] = wert then
          index = i
          break
      end if
  ende für
  if index = -1 then
      return
  end if
  für j von index bis n - 2
      a[j] = a[j + 1]
  ende für
  // n = n - 1
endprozedur
```

در نسخه‌های شرطی، قبل از درج یا حذف، یک شرط چک می‌شود: مثلاً وجود فضای خالی، یا این‌که آیتم مورد نظر واقعاً در آرایه وجود داشته باشد. این سبک در امتحان‌ها خیلی رایج است (به‌خصوص «حذف اولین وقوع یک مقدار»).

5. Generische Varianten mit Function<T> – نسخه‌های جنریک

5.1 Index eines Elements finden mit Function<T>

```
function findeIndex(liste: Array von T, bedingung: Function<T>): Integer
  für i von 0 bis länge(liste) - 1
      if bedingung(liste[i]) then
          return i
      end if
  ende für
  return -1
end function
```

5.2 Entfernen eines Elements nach Bedingung

```
prozedur entferneNachBedingung(liste: Array von T, n: Integer, bedingung:
                                                    Function<T>)
  index = findeIndex(liste, bedingung)
  if index = -1 then
      return
  end if
  für i von index bis n - 2
      liste[i] = liste[i + 1]
  ende für
  // n = n - 1
endprozedur
```

در نسخهٔ جنریک با **<Function<T** ابتدا با استفاده از تابع **bedingung** اندیس مورد نظر پیدا می‌شود و سپس با همان الگوی شیفت به چپ، عنصر حذف می‌شود.

6. Typische Fehler – خطاهای رایج در Einfügen/Entfernen

6.1 Falsche Richtung beim Verschieben

```
prozedur fuegeEinFehler(a: Array von Integer, n: Integer, pos: Integer, wert: Integer
  // FEHLER: falsche Schleifenrichtung
  für i von pos bis n - 1
      a[i + 1] = a[i]
  ende für
  a[pos] = wert
endprozedur
```

Fehler: Hier werden die Elemente von links nach rechts überschrieben, so dass Werte verloren gehen.

6.2 Entfernen ohne vollständiges Verschieben

```
prozedur entferneFehler(a: Array von Integer, n: Integer, pos: Integer)
  // FEHLER: nur das Element auf pos wird "gelöscht"
  a[pos] = 0
  // aber die restlichen Elemente werden nicht verschoben
endprozedur
```

6.3 Falsche Grenzen (off-by-one)

```
prozedur entferneAnPosFehler(a: Array von Integer, n: Integer, pos: Integer)
  für i von pos bis n - 1     // FEHLER: sollte bis n - 2 laufen
      a[i] = a[i + 1]
  ende für
endprozedur
```

خطاهای رایج در این نوع الگوریتم‌ها:

- شیفت را از جهت اشتباه انجام می‌دهند (مثلاً از چپ به راست به‌جای از راست به چپ).
- فقط خانهٔ موردنظر را صفر یا خالی می‌کنند، بدون این‌که بقیهٔ عناصر را شیفت دهند.
- حلقه را تا اندیس اشتباه اجرا می‌کنند (مثلاً تا `n - 1` به‌جای `n - 2`).

7. Prüfungsähnliche Übungen – تمرین‌های شبیه امتحان

Übung 1 – Schüler an Position einfügen

```
  prozedur fuegeSchuelerEin(schueler: Array von Schueler, n: Integer, pos:
                                          Integer, neu: Schueler)
    // Implementieren Sie das Einfügen mit Rechts-Shift.
endprozedur
```

Übung 2 – Artikel mit bestimmter Artikelnummer entfernen

```
  prozedur entferneArtikelMitNummer(artikel: Array von Artikel, n:
                                          Integer, nummer: Integer)
    // 1. Index des Artikels mit artikel[i].nummer = nummer finden
    // 2. Wenn gefunden, alle Elemente ab diesem Index nach links schieben
endprozedur
```

Übung 3 – Einfügen nur, wenn Platz frei ist

```
  prozedur fuegeRaumEinWennPlatz(raeume: Array von Raum, n: Integer, maxGroesse:
                                                  Integer, neu: Raum)
    // Einfügen am Ende, aber nur wenn n < maxGroesse
endprozedur
```

Übung 4 – Generisch entfernen mit Function<T>

```
  prozedur entferneNachBedingung(liste: Array von T, n: Integer, bedingung:
                                                  Function<T>)
   // Verwenden Sie findeIndex(liste, bedingung) und verschieben Sie danach alle
  Elemente nach links. endprozedur
```

Übung 5 – Fehleranalyse

```
  prozedur fuegeEinFehler2(a: Array von Integer, n: Integer, pos: Integer, wert: Intege
    für i von 0 bis n - 1
        a[i + 1] = a[i]
    ende für
    a[pos] = wert
endprozedur
```

Aufgabe: Erklären Sie, warum die Funktion `fuegeEinFehler2` falsch ist, und schreiben Sie eine korrigierte Version.

وظیفه: توضیح بده چرا تابع **`fuegeEinFehler2`** اشتباه است (همه‌چیز را از ۰ شیفت می‌دهد و داده‌ها را خراب می‌کند) و نسخهٔ درست آن را بنویس.

Ende der Masterdatei – Einfügen & Entfernen im Array.

پایان فایل مستر Einfügen & Entfernen im Array.