

Aufgabe 2: Schwimmbad

Team: BANDO

Einsendenummer: 00222

1. November 2017

Inhaltsverzeichnis

1 Lösungsidee	1
2 Umsetzung	2
3 Beispiele	2
3.1 Beispiel aus der Angabe	2
3.2 Beispiel1.txt	2
3.3 Beispiel2.txt	3
3.4 Beispiel3.txt	4
4 Quellcode	4

1 Lösungsidee

Die Unübersichtlichkeit der Eintrittspreise legt einen systematischen Brute-Force-Algorithmus als Lösung nahe. Das Programm betrachtet jedoch zuerst die Eintrittsmöglichkeit der Vierjährigen, was ohne Umstände gleich am Anfang bestimmt werden kann.

Es wird zunächst ein 6-Tupel definiert, der einen möglichen Ticketkauf bei einem Eintritt beschreibt, und wie folgt aussieht:

$$E := (O, U, T, F, P, G)$$

O steht für die Anzahl der Einzelkarten für die o16er, U für die Anzahl der Einzelkarten für die u16er, T für die Anzahl der Tageskarten, F für die Anzahl der Familienkarten, P für die Anzahl der für den Rabatt verwendeten Gutscheine und G für die Anzahl der mitgeführten Gutscheine.

Als erstes werden für alle Werte des 6-Tupels gültige Obergrenzen gesucht, sodass gilt, dass im jeden Fall jeder andere Ticketkauf mit einer Überschreitung dieser Obergrenzen nicht der günstigste ist:

- $O_{max} :=$ Anzahl der o16er
- $U_{max} :=$ Anzahl der u16er
- $T_{max} := \lfloor (\text{Anzahl der o16er} + \text{Anzahl der u16er}) / 6 \rfloor$
- $F_{max} := \max(\lfloor \text{Anzahl der o16er} / 2 \rfloor; \lfloor \text{Anzahl der u16er} / 2 \rfloor)$
- $P_{max} := 1$

An einem Wochenende sind keine Tageskarten gestattet, womit $T_{max} = 0$ gilt. Analog gilt in den Ferien $P_{max} = 0$ und $G = 0$ (denn schließlich kann man die Gutscheine auch gleich zu Hause lassen).

Nun sollen die zwei Methoden *gueltigkeit*(E) und *preis*(E) hinzugenommen werden. Die erste Methode bestimmt in Abhängigkeit von der zu wünschen eintretenden Gruppe, ob mit den erworbenen Tickets ein Eintritt möglich ist. Die zweite Methode errechnet den damit verbundenen Preis.

Mit Hilfe der vordefinierten Obergrenzen können nun eine endliche Zahl möglicher Ticketeinkäufe durchiteriert werden, unter denen der günstigste Ticketeinkauf dabei ist. Von jedem gültigen Ticketkauf soll der Preis errechnet werden. Der billigste Ticketkauf kann nun bestimmt werden.

2 Umsetzung

Die Methode $preis(E)$ ist sehr leicht umzusetzen: Die erworbenen Karten werden mit ihrem jeweiligem Preis, der vom Wochentag abhängt, multipliziert und anschließend addiert. Wird ein Rabattgutschein verwendet, so wird das Endergebnis zusätzlich um ein Zehntel reduziert.

Für die Methode $gueltigkeit(E)$ jedoch durchläuft man alle Möglichkeiten eine Familienkarte zu nutzen, also 2x o16er und 2x u16er, oder 1x o16 und 3x u16er. Danach speichert man die Anzahl der o16er und 16er als temporäre Zwischenvariablen und geht wie folgt vor: Man reduziert die Anzahl der o16er um O und um die durch die Familienkarten gestatteten o16er, analog verringere man die Anzahl der u16er (für auftretende negative Anzahlen runde man auf 0 auf). Die Summe der reduzierten Anzahlen der o16er und u16er ist die Zahl der Personen, die noch kein Ticket haben. Zuletzt wird überprüft, ob diese Personen alle gemeinsam mit Hilfe der übrigen Tageskarten und Gutscheine mit reingenommen werden können. Trifft dies in mindestens einem Fall zu, so ist das 6-Tupel E gültig.

3 Beispiele

3.1 Beispiel aus der Angabe

Anzahl
 unter 4 jährigen: 1
 4 bis 16 jährigen: 4
 über 16 jährigen: 1
 Gutscheine: 1
 Ist Wochenende: 1
 Ist Ferien: 1

Zu kaufende Tickets:
 Einzelkarten Ueber 16: 0
 Einzelkarten 4 bis 16: 1
 Tageskarten: 0
 Familientickets: 1
 10Prozent Gutschein: 0
 Ticket-Gutscheine: 0

Das macht einen Gesamtpreis von 10.5 Euro

Das Ergebnis scheint richtig, da in den Ferien keine Gutscheine verwendet werden dürfen und ein Familienticket mit Einzelkarte günstiger als lauter Einzelkarten sind. Tageskarten sind am Wochenende schließlich nicht gestattet.

3.2 Beispiel1.txt

Anzahl
 unter 4 jährigen: 0
 4 bis 16 jährigen: 0
 über 16 jährigen: 27
 Gutscheine: 3
 Ist Wochenende: 0
 Ist Ferien: 1

Zu kaufende Tickets:
 Einzelkarten Ueber 16: 3
 Einzelkarten 4 bis 16: 0
 Tageskarten: 4
 Familientickets: 0
 10Prozent Gutschein: 0
 Ticket-Gutscheine: 0

Aufgabe 2: Schwimmbad

Das macht einen Gesamtpreis von 52.4 Euro

Das Ergebnis scheint richtig, da eine Tageskarte billiger als sechs Einzelkarten sind. Familientickets rentieren sich selbstverständlich gar nicht, da keine unter 16 Jährigen mitgenommen werden. Gutscheine sind ebenfalls nicht in den Ferien erlaubt.

Anzahl
unter 4 jährigen: 0
4 bis 16 jährigen: 0
über 16 jährigen: 27
Gutscheine: 3
Ist Wochenende: 1
Ist Ferien: 0

Zu kaufende Tickets:
Einzelkarten Ueber 16: 25
Einzelkarten 4 bis 16: 0
Tageskarten: 0
Familientickets: 0
10Prozent Gutschein: 1
Ticket-Gutscheine: 2

Das macht einen Gesamtpreis von 78.75 Euro

Im Gegensatz zum obigen Fall dürfen Gutscheine zwar verwendet werden, jedoch keine Tageskarten. Unter der Woche in den Ferien ist ein Besuch daher deutlich günstiger.

3.3 Beispiel2.txt

Anzahl
unter 4 jährigen: 1
4 bis 16 jährigen: 4
über 16 jährigen: 5
Gutscheine: 0
Ist Wochenende: 0
Ist Ferien: 1

Zu kaufende Tickets:

Einzelkarten Ueber 16: 0
Einzelkarten 4 bis 16: 3
Tageskarten: 1
Familientickets: 0
10Prozent Gutschein: 0
Ticket-Gutscheine: 0

Das macht einen Gesamtpreis von 17.0 Euro

Für die Großfamilie tun sich zwei Möglichkeiten auf: Zwei Familienkarten oder eine Tageskarte, wobei in beiden Fällen zusätzlich Einzeltickets benötigt werden. Für den ersten Fall kostet es insgesamt $2 \cdot 8 + 1 \cdot 2.8$ Euro = 18.8 Euro, während es im zweiten Fall $1 \cdot 11 + 3 \cdot 2$ Euro = 17 Euro kostet. Unser Programm behält wie erwartet Recht.

Anzahl
unter 4 jährigen: 1
4 bis 16 jährigen: 4
über 16 jährigen: 5
Gutscheine: 0
Ist Wochenende: 1

Ist Ferien: 1

—

Zu kaufende Tickets:

Einzelkarten Ueber 16: 1

Einzelkarten 4 bis 16: 0

Tageskarten: 0

Familientickets: 2

10Prozent Gutscheine: 0

Ticket-Gutscheine: 0

—

Das macht einen Gesamtpreis von 19.5 Euro

Am Wochenende kann die Großfamilie keine Tageskarten erwerben. Hinzu kommt, dass die Einzeltickets nun auch teurer sind. Die Familientickets scheinen nun in diesem Fall sehr rentabel. Die Großfamilie sollte aber dennoch Werktags das Schwimmbad besuchen.

3.4 Beispiel3.txt

Anzahl

unter 4 jährigen: 0

4 bis 16 jährigen: 0

über 16 jährigen: 14

Gutscheine: 1

Ist Wochenende: 1

Ist Ferien: 0

—

Zu kaufende Tickets:

Einzelkarten Ueber 16: 14

Einzelkarten 4 bis 16: 0

Tageskarten: 0

Familientickets: 0

10Prozent Gutscheine: 1

Ticket-Gutscheine: 0

—

Das macht einen Gesamtpreis von 44.1 Euro

Das Ergebnis scheint richtig, da weder Tageskarten verwendet werden dürfen, noch Familienkarten rentabel sind. Der Gutschein wird ersichtlicherweise für den 10 Prozent Rabatt verwendet werden.

Anzahl

unter 4 jährigen: 0

4 bis 16 jährigen: 0

über 16 jährigen: 18

Gutscheine: 1

Ist Wochenende: 1

Ist Ferien: 0

—

Zu kaufende Tickets:

Einzelkarten Ueber 16: 18

Einzelkarten 4 bis 16: 0

Tageskarten: 0

Familientickets: 0

10Prozent Gutscheine: 1

Ticket-Gutscheine: 0

—

Das macht einen Gesamtpreis von 56.7 Euro

Analog gilt auch die Richtigkeit des Ergebnisses für den anderen Kegelverein. Der Gesamtpreis des

Eintritts für die beiden Kegelvereine einzeln beträgt $44.1 \text{ Euro} + 56.7 \text{ Euro} = 100.8 \text{ Euro}$

Anzahl
 unter 4 jährigen: 0
 4 bis 16 jährigen: 0
 über 16 jährigen: 32
 Gutscheine: 2
 Ist Wochenende: 1
 Ist Ferien: 0

Zu kaufende Tickets:
 Einzelkarten Ueber 16: 31
 Einzelkarten 4 bis 16: 0
 Tageskarten: 0
 Familientickets: 0
 10Prozent Gutschein: 1
 Ticket-Gutscheine: 1

Das macht einen Gesamtpreis von 97.65 Euro

Auch in diesem Fall bleibt den Kegelvereinen nichts anderes als das Erwerben lauter Einzelkarten. Nun sparen sie sich jedoch durch den zusätzlichen Gutschein den Eintritt einer einzigen Person. Die Menge an erspartem Geld beträgt somit $100.8 \text{ Euro} - 97.65 \text{ Euro} = 3.15 \text{ Euro}$.

4 Quellcode

Der folgende Auszug zeigt die Umsetzung der Methoden *gueltigkeit(E)* und *preis(E)*, sowie den Teil des Programms, das alle möglichen Ticketkäufe durchiteriert:

```

1  # Prueft, ob die Zusammenstellung von Tickets gueltig ist
   def gueltigkeit(o, u, t, f, p, g):
3      # Wenn ein Gutschein fuer den 10Prozent-Rabatt-Bonus gebraucht wird,
      # ohne dass ein Gutschein da ist,
5      if p > g:
          # Dann gib False zurueck
7          return False
      for i in range(f + 1):
9          A = i # Anzahl Familienkarten Typ A: 2 o16 + 2 u16
          B = f - i # Anzahl Familienkarten Typ B: 1 o16 + 3 u16
11         # Anzahl der u16er, die durch die Familienkarten
          # mitgenommen werden koennen
13         fu = 2 * A + 3 * B
          # Anzahl der o16er, die durch die Familienkarten
15         # mitgenommen werden koennen
          fo = 2 * A + 1 * B
17         # Anzahl der u16er, die nicht durch Einzel- oder
          # Familienkarte reinkommen
19         zu = u16 - u - fu
          # Anzahl der o16er, die nicht durch Einzel- oder
21         # Familienkarte reinkommen
          zo = o16 - o - fo
23         # Wenn zu viele u16er Tickets da sind,
          if zu < 0:
25             # Dann sind alle u16er drin
             zu = 0
27         # Wenn zu viele o16er Tickets da sind,
          if zo < 0:

```

```

29         # Dann sind alle o16er drin
        zo = 0
31         # Anzahl der noch unterzubringenden Personen
        zges = zu + zo
33         # Wenn diese Anzahl kleiner ist als die noch zur
        # Verfuegung stehenden Freikarten
35         if zges <= 6 * t + g - p:
            # Dann gib True zurueck
37             return True
        # Gib False zurueck
39         return False

41 # Berechnet aus der gegebenen Kombination von Tickets den Gesamtpreis
def preis(o, u, t, f, p, g):
43     # Wenn der Termin am Wochenende ist,
    if wo:
45         # Dann Berechnung des Preises mit dem Normal-Preisen
        preisges = o * 3.5 + u * 2.5 + t * 11 + f * 8
47     else: # Sonst:
        # Berechnung des Preises mit den -20Prozent-Preisen
49         preisges = o * 2.8 + u * 2 + t * 11 + f * 8
        # Wenn p gleich 1 ist bzw. Wenn ein
        # 10Prozent-Gutschein verwendet wird:
51         if p:
            # Dann ziehe vom Gesamtpreis 10 Prozent ab
            preisges -= preisges * 0.1
53         # Gebe den Gesamtpreis zurueck
        return preisges
55
57 # Minimaler Preis
mpreis = -1
59 # Als Liste realisierter 6-Tupel des billigsten Ticketeinkaufs
meintritt = []
61 for O in range(Omax + 1): # Durchlaufe 0 bis Omax als O
    for U in range(Umax + 1): # Durchlaufe 0 bis Umax als U
        for T in range(Tmax + 1): # Durchlaufe 0 bis Tmax als T
            for F in range(Fmax + 1): # Durchlaufe 0 bis Fmax als F
                for P in range(Pmax + 1): # Durchlaufe 0 bis Pmax als P
63                     # Wenn diese Kombination von Tickets gueltig ist,
                    if gueltigkeit(O, U, T, F, P, G):
65                         # Dann berechne den Preis dieser Kombination
                        zpreis = preis(O, U, T, F, P, G)
67                         # Wenn dieser Preis kleiner als der
                        # bisher billigste Preis ist:
                        if zpreis < mpreis or mpreis == -1:
69                             # Dann ueberschreibe den billigsten
                                # Preis mit dem eben berechneten
                                mpreis = zpreis
71                             # Ueberschreibe die Liste mit dem billigerem 6-Tupel
                                meintritt = [O, U, T, F, P, u16+o16-O-U-5*T-4*F]
73
75
77
79

```