Aufgabenblatt 3

Florian Ludewig (Übungsgruppe 2)

November 12, 2019

Aufgabe 1

a)

Ausdruck	Wert
`int x = 49;`	–
`!(x >= 0 && x < 50 || x > 50 && x <= 100)`	`false`
`x++ == 49 || x-- == 49`	`true`
`!(x >= 0 && x < 50 || x > 50 && x <= 100)`	`true`
`int k; scanf("%d", &k);`	–
`!(k > 0 && k%10 <= 7) == (!(k > 0) || !(k%10 <= 7))` $\neg(k > 0 \wedge k (mod\ 10) \leq 7) = (\neg(k > 0) \vee \neg(k (mod\ 10) \leq 7))$ Definiere $P := k > 0$ und $Q := k (mod\ 10) \leq 7$. Dann gilt: $\neg(P \wedge Q) \Leftrightarrow \neg P \vee \neg Q$, weil es laut den De Morganschen Gesetzen äquivalent ist.	`true`

b)

```
1) !(m > 10) && (n != 0)

2) (x > 0) && (y > 0) && (z > 0) && (x % 5 == 0) && (y % 5 == 0) && (z % 5
    == 0)

3) (a >= 1 && a <= 10) || (a < -7)

4) 1 < 3
```

Aufgabe 2

```c
#include <stdio.h>
#include <stdbool.h>
#include <stdlib.h>

struct station
{
  int line;
  int stop;
};

struct trip
{
  struct station start;
  struct station end;
};

bool validate_input(struct trip trip, int MIN, int MAX)
{
  if (trip.start.line > MAX || trip.start.stop > MAX ||
      trip.end.line > MAX || trip.end.stop > MAX ||
      trip.start.line < MIN || trip.start.stop < MIN ||
      trip.end.line < MIN || trip.end.stop < MIN)
    return false;
  return true;
}

bool is_short_trip(struct trip trip)
{
  if ((trip.start.line == 5 && trip.start.stop == 2) &&
          (trip.end.line == 4 && trip.end.stop == 2) ||
      (trip.start.line == 4 && trip.start.stop == 2) &&
          (trip.end.line == 5 && trip.end.stop == 2))
    return false;

  if ((trip.start.line == 0 && trip.start.stop == 0) &&
          (trip.end.line == 2 && trip.end.stop == 1) ||
      (trip.start.line == 2 && trip.start.stop == 1) &&
          (trip.end.line == 0 && trip.end.stop == 0))
    return false;

  int stop_difference = abs(trip.start.stop - trip.end.stop);
  if (stop_difference == 1)
  {
    if (trip.start.line == trip.end.line)
      return true;
    if (trip.start.stop == 0 && trip.end.stop == 1 ||
        trip.start.stop == 1 && trip.end.stop == 0)
      return true;
  }

  return false;
}

int count_zone_crosses(struct trip trip)
{
  if (trip.start.stop > 2 && trip.end.stop > 2 &&
      trip.start.line != trip.end.line)
    return 2;
  if (trip.start.stop > 2 && trip.end.stop < 3)
    return 1;
  if (trip.start.stop < 3 && trip.end.stop > 2)
    return 1;
  return 0;
```

```
64  }
65
66  int count_end_stops(struct trip trip)
67  {
68    int end_stops = 0;
69    if (trip.start.stop == 5)
70      end_stops++;
71    if (trip.end.stop == 5)
72      end_stops++;
73    return end_stops;
74  }
75
76  int main()
77  {
78    int start_input, end_input;
79    const int DEFAULT_PRICE = 3,
80              SHORT_TRIP_PRICE = 2,
81              END_STOP_PRICE = 1,
82              ZONE_CROSS_PRICE = 1;
83
84    printf("Starthaltestelle: ");
85    scanf("%d", &start_input);
86    printf("Zielhaltestelle: ");
87    scanf("%d", &end_input);
88
89    struct station start_station = {(start_input / 10) % 10, start_input %
          10};
90    struct station end_station = {(end_input / 10) % 10, end_input % 10};
91    struct trip trip = {start_station, end_station};
92    if (!validate_input(trip, 0, 5))
93    {
94      printf("Ungueltige Eingabe\n");
95      return 1;
96    }
97
98    int price = 0;
99    if (!(trip.start.line == trip.end.line && trip.start.stop == trip.end.stop
          ))
100   {
101     price = is_short_trip(trip) ? SHORT_TRIP_PRICE : DEFAULT_PRICE;
102     price += count_zone_crosses(trip) * ZONE_CROSS_PRICE;
103     price += count_end_stops(trip) * END_STOP_PRICE;
104   }
105
106   printf("%d Euro\n", price);
107   return 0;
108 }
```