



Schweizer Informatik-Olympiade



Nachwuchsförderung durch
kompetitive Programmierwettbewerbe



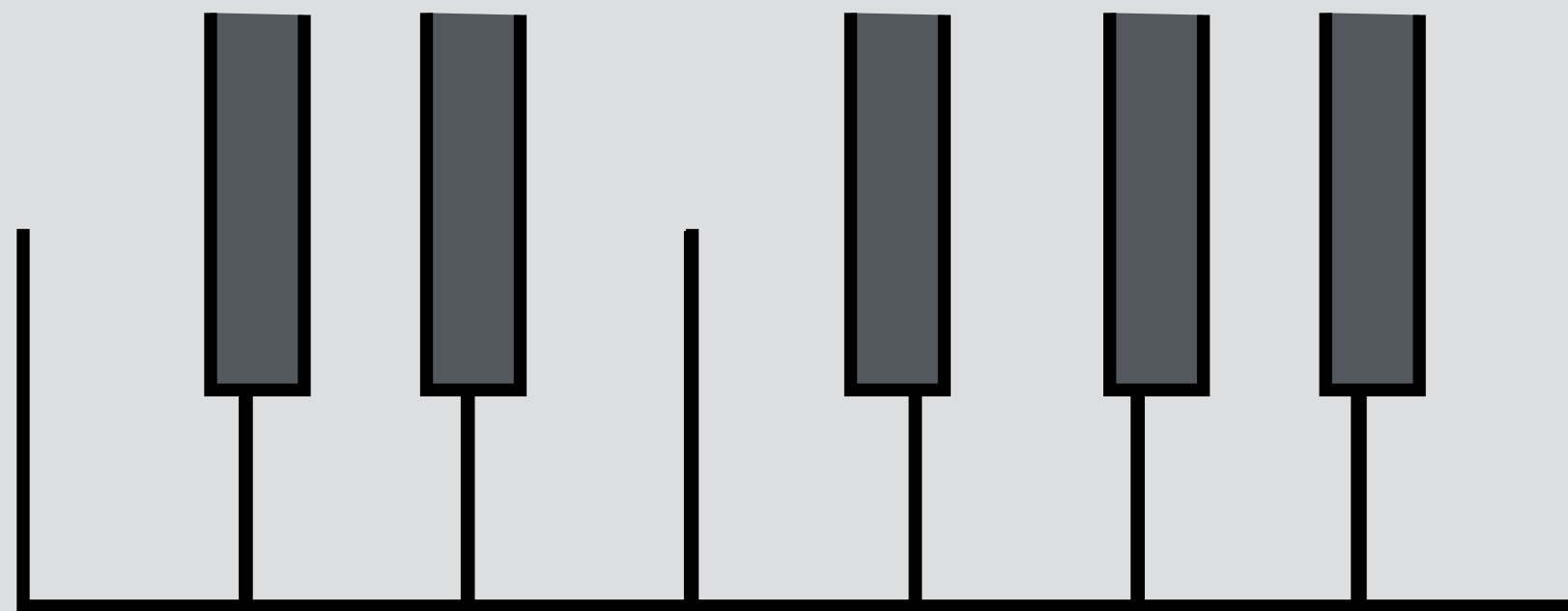
SOI STIU Contest

<http://soi.ch/stiu/index.html>

- 3 Aufgaben zur dynamischen Programmierung
- Vorlagen in den Programmiersprachen C++, Java und Python
- alle Slides und Links zum Vortrag

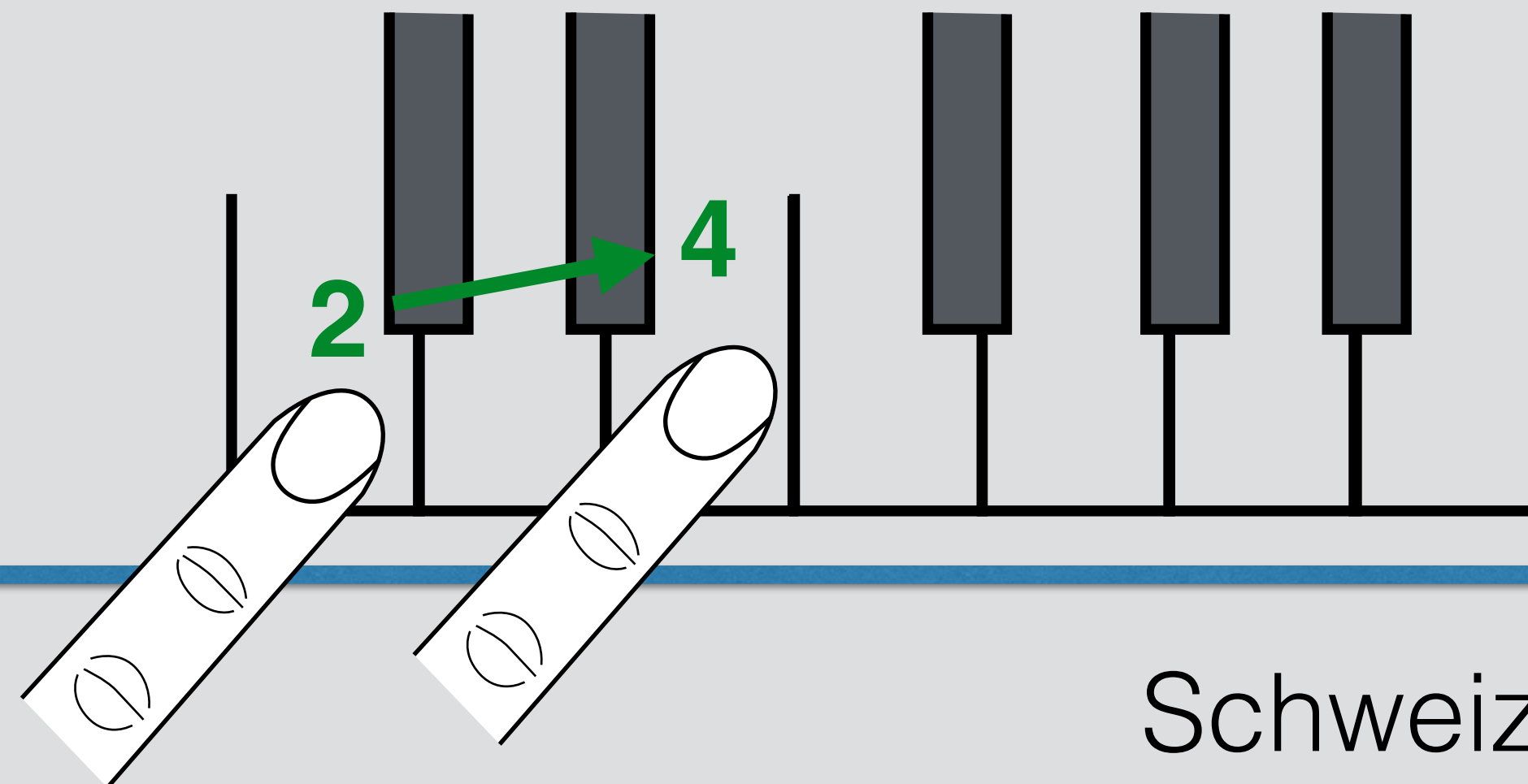
Beispiellösung Piano

- Aufgabe:
- Gegeben: Ein einstimmiges Lied durch Tastenzahlen
z.B.: 1,3,2,4,6,2,7,4,2
- Gesucht: Fingersatz für eine Hand mit maximaler Eleganz



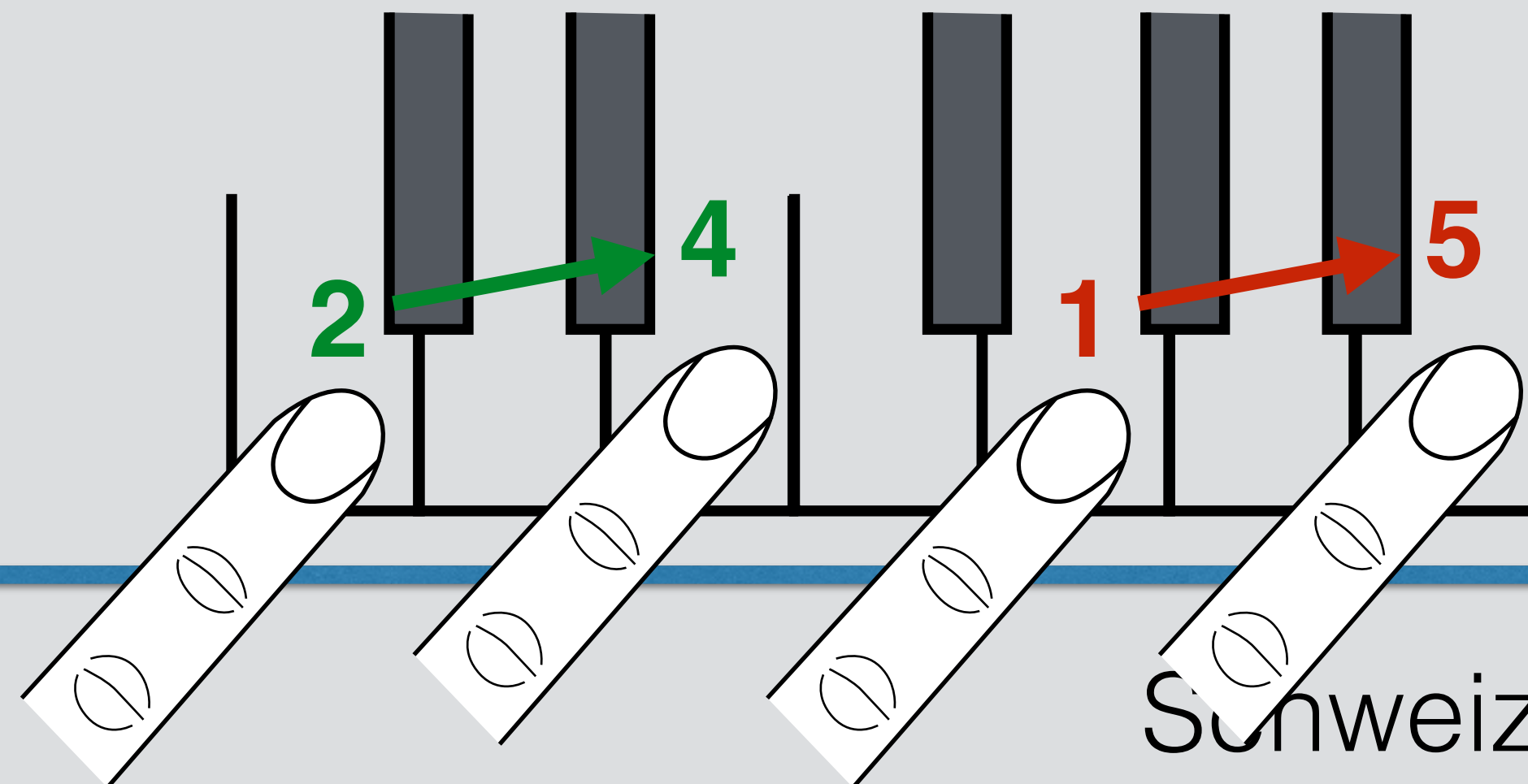
Beispiellösung Piano

- Aufgabe:
- Gegeben: Ein einstimmiges Lied durch Tastenzahlen
z.B.: 1,3,2,4,6,2,7,4,2
- Gesucht: Fingersatz für eine Hand mit maximaler Eleganz



Beispiellösung Piano

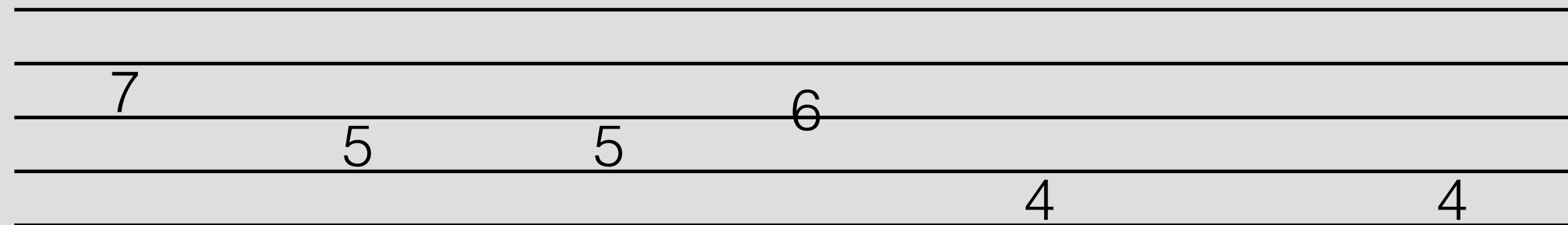
- Aufgabe:
- Gegeben: Ein einstimmiges Lied durch Tastenzahlen
z.B.: 1,3,2,4,6,2,7,4,2
- Gesucht: Fingersatz für eine Hand mit maximaler Eleganz



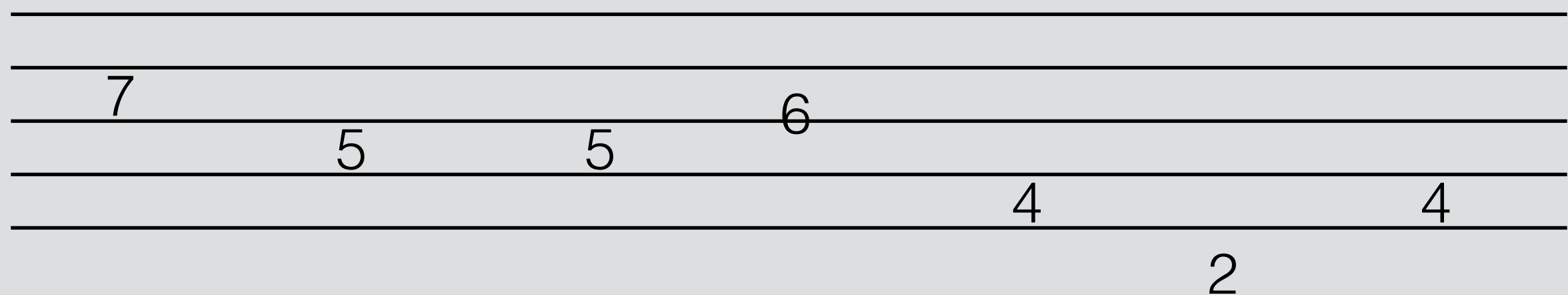
Beispiellösung Piano

- Schlüsselbeobachtung: Eleganz einer Note hängt nur von zwei Tastenzahlen und den dafür gewählten Fingern ab
 - elegant falls $t_i - t_{i-1} = f_i - f_{i-1}$
- DP-Zustand: maximale Eleganz für die ersten i Noten mit bestimmten letzten Finger j :
 - $DP[i][j]$ für Noten $1 \leq i \leq n$ und Finger $1 \leq j \leq 5$

Beispiellösung Piano

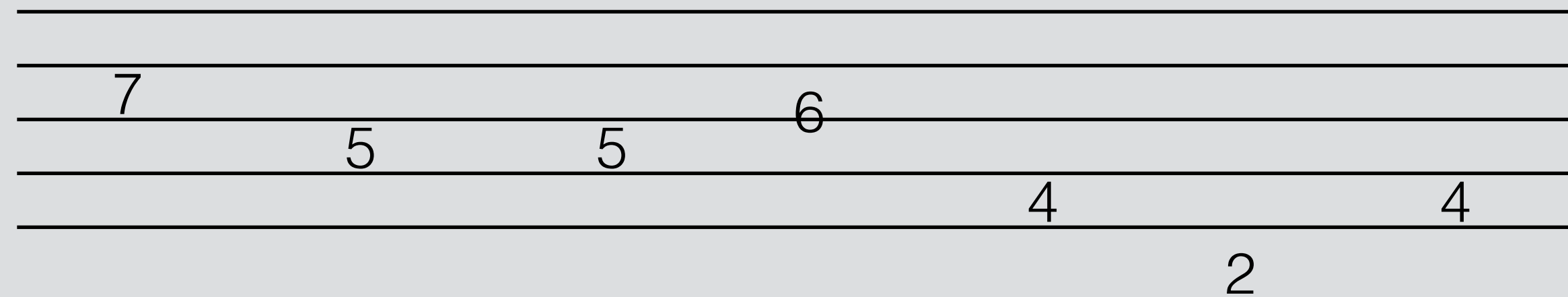


Beispiellösung Piano



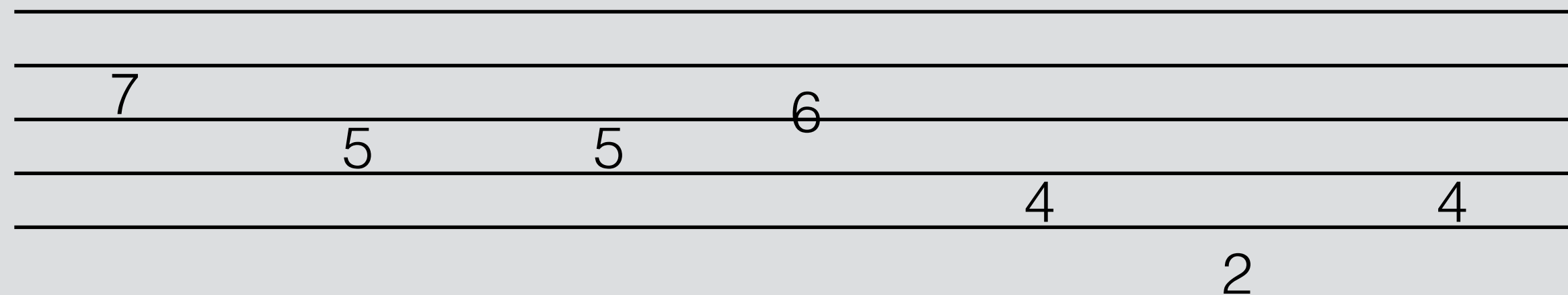
		7	5	5	6	4	2	4	Taste
Finger	1	1							
	2	1							
	3	1							
	4	1							
	5	1							

Beispiellösung Piano



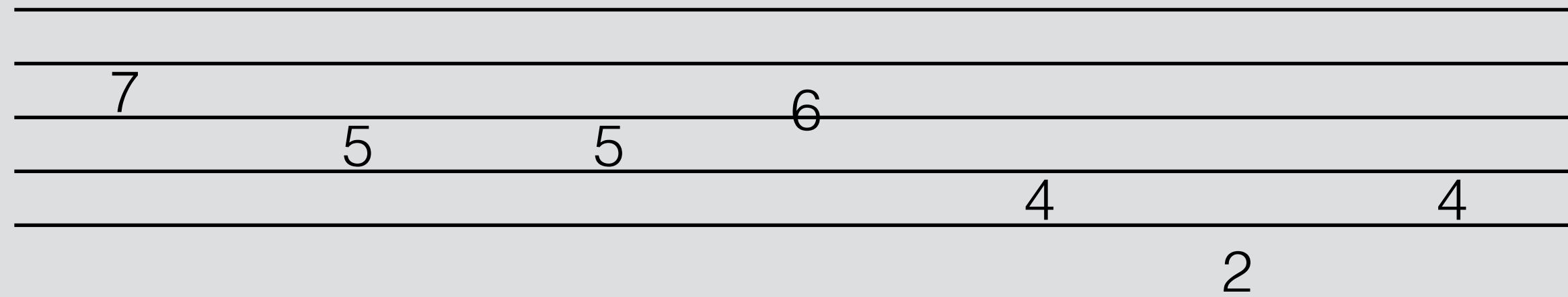
		7	5	5	6	4	2	4	Taste
Finger	1	1	2						
	2	1							
	3	1							
	4	1							
	5	1							

Beispiellösung Piano



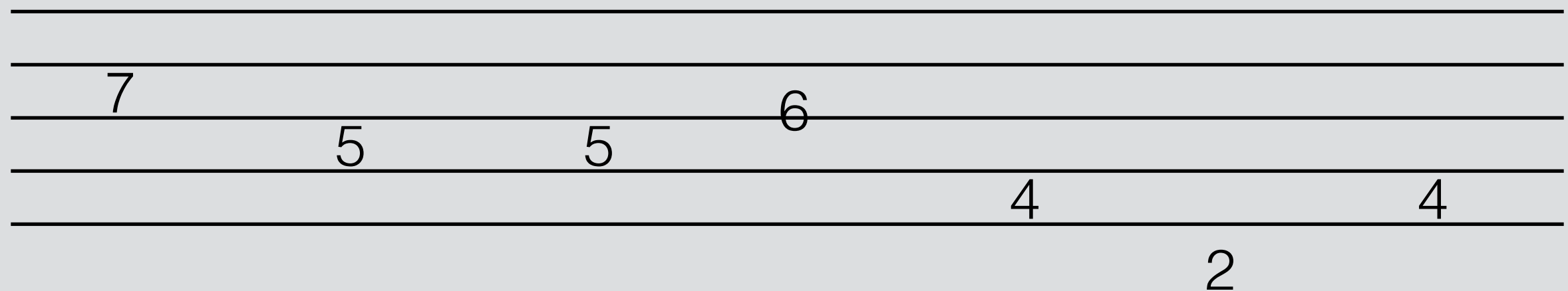
		7	5	5	6	4	2	4	Taste
Finger	1	1	2						
	2	1	2						
	3	1							
	4	1							
	5	1							

Beispiellösung Piano



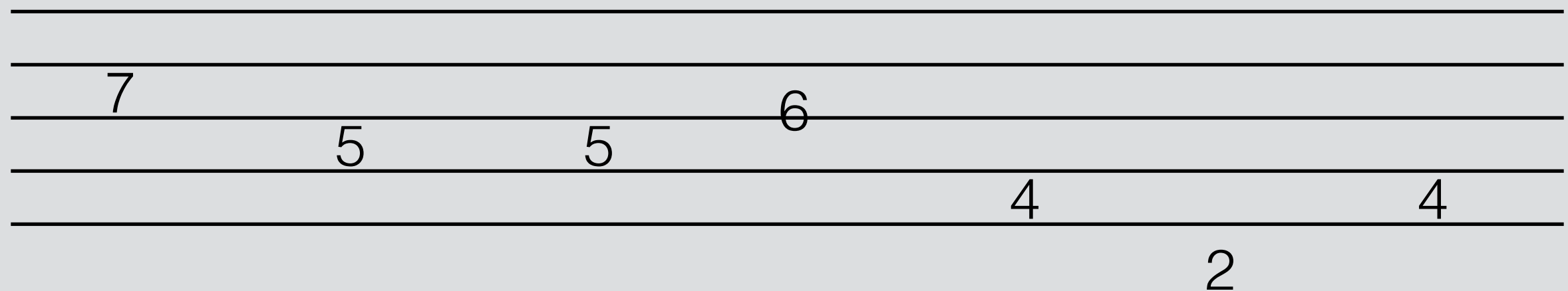
Finger	7	5	5	6	4	2	4	Taste
	1	2						
	1	2						
	1	2						
	1							
	1							

Beispiellösung Piano



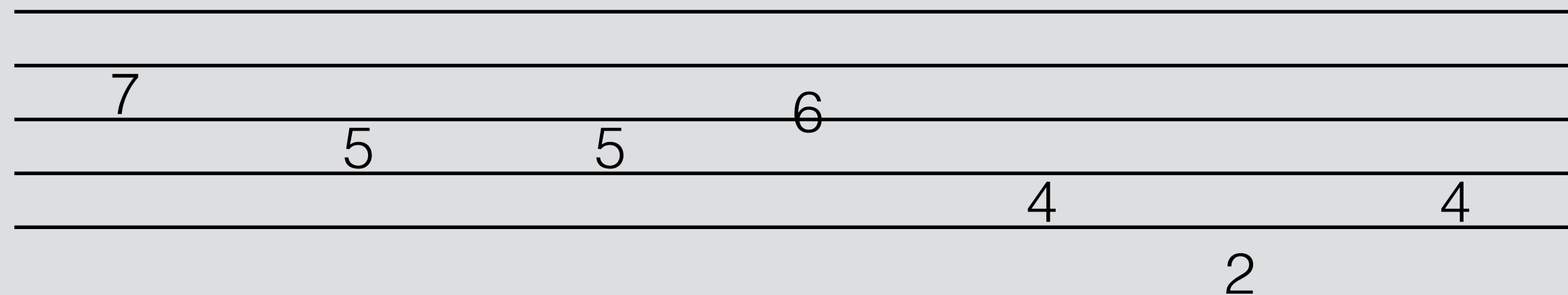
		7	5	5	6	4	2	4	Taste
Finger	1	1	2						
	2	1	2						
	3	1	2						
	4	1	1						
	5	1	1						

Beispiellösung Piano



		7	5	5	6	4	2	4	Taste
Finger	1	1	2	← 3					
	2	1	2	← 3					
	3	1	2	← 3					
	4	1	1	← 2					
	5	1	1	← 2					

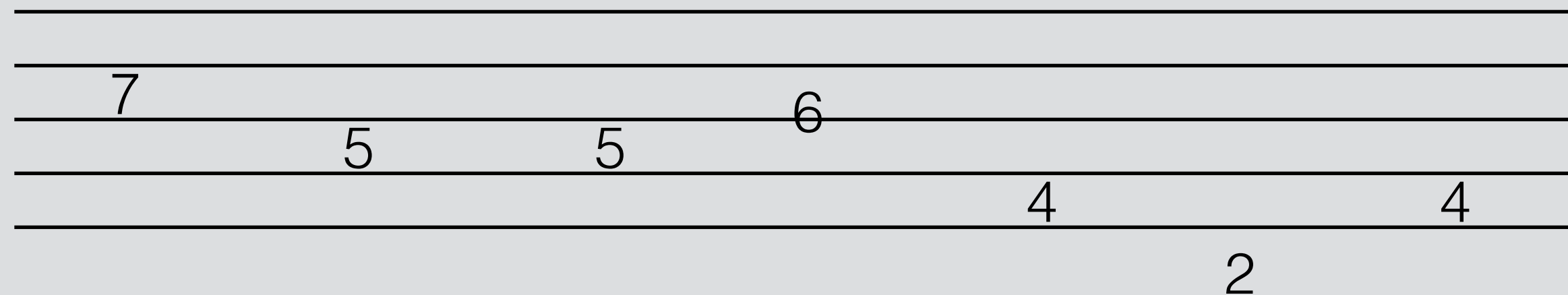
Beispiellösung Piano



		7	5	5	6	4	2	4	Taste
		1	2	3	3				
Finger	1	1	2	3	3				
	2	1	2	3	4				
	3	1	2	3	4				
	4	1	1	2	4				
	5	1	1	2	3				

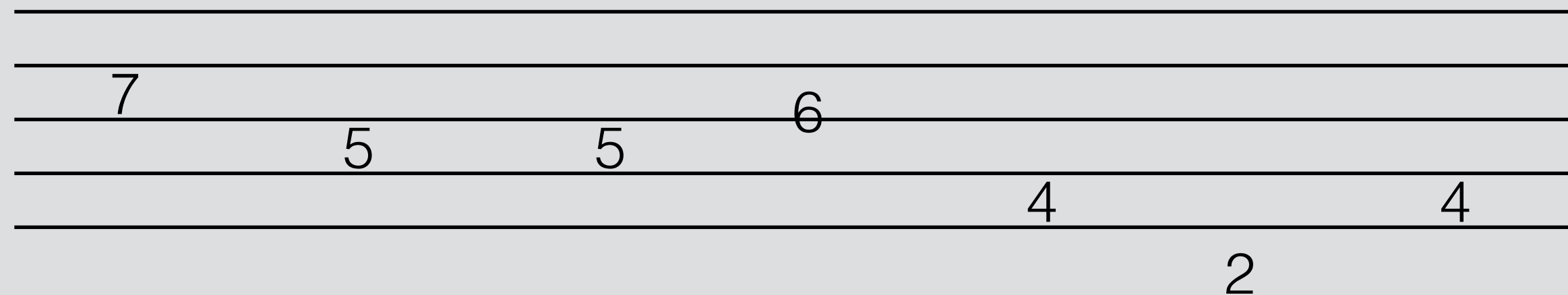
Four green arrows indicate finger transitions between notes in the sequence: from note 3 to note 3, from note 3 to note 4, from note 4 to note 4, and from note 2 to note 3.

Beispiellösung Piano



		7	5	5	6	4	2	4	Taste
Finger	1	1	2	3	3	5			
	2	1	2	3	4	5			
	3	1	2	3	4	4			
	4	1	1	2	4	4			
	5	1	1	2	3	4			

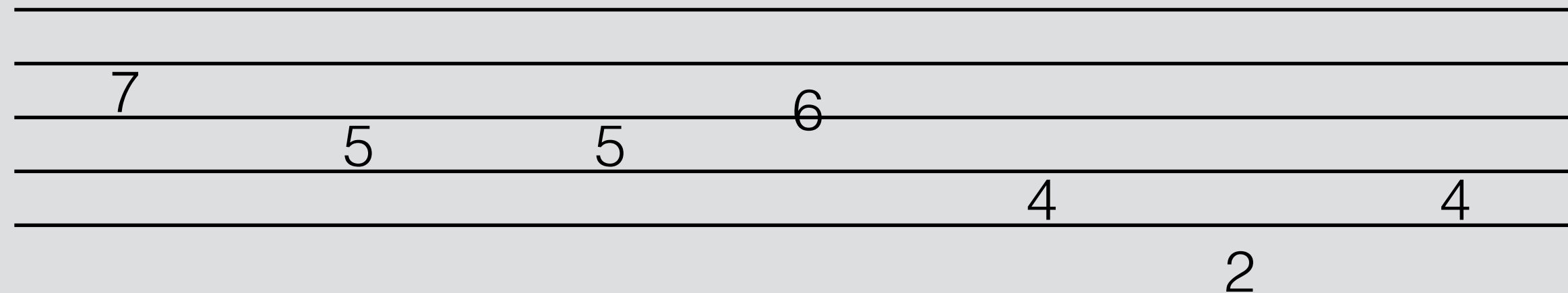
Beispiellösung Piano



		7	5	5	6	4	2	4	Taste
Finger	1	1	2	3	3	5	5		
	2	1	2	3	4	5	5		
	3	1	2	3	4	4	5		
	4	1	1	2	4	4	5		
	5	1	1	2	3	4	5		

Three green arrows pointing from the 'Taste' column to the 'Finger' column. The first arrow points from the '5' in the first row to the '1' in the first row. The second arrow points from the '5' in the second row to the '2' in the second row. The third arrow points from the '5' in the third row to the '3' in the third row.

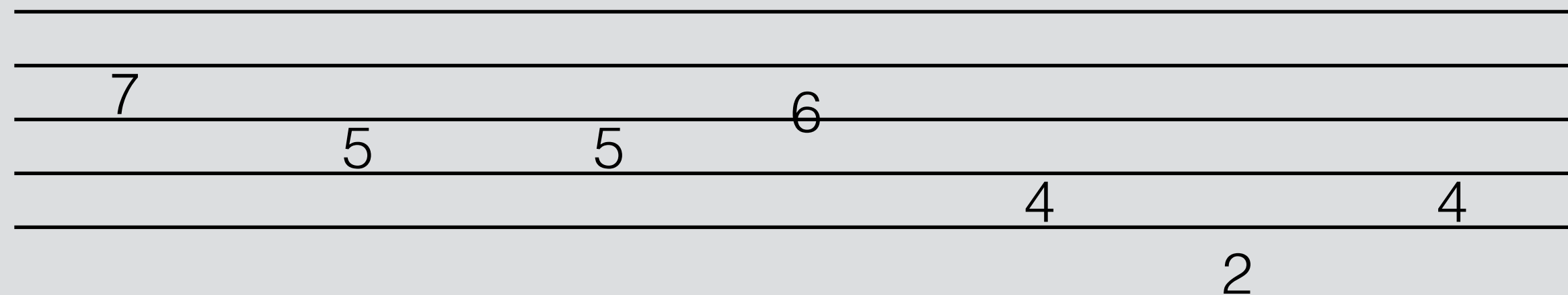
Beispiellösung Piano



		7	5	5	6	4	2	4	Taste
Finger	1	1	2	3	3	5	5	5	
	2	1	2	3	4	5	5	5	
	3	1	2	3	4	4	5	6	
	4	1	1	2	4	4	5	6	
	5	1	1	2	3	4	5	6	

Three green arrows pointing from the 'Taste' column to the 'Finger' column. The first arrow points from the '5' in the 'Taste' column of the first row to the '5' in the 'Finger' column of the first row. The second arrow points from the '5' in the 'Taste' column of the second row to the '5' in the 'Finger' column of the second row. The third arrow points from the '6' in the 'Taste' column of the third row to the '5' in the 'Finger' column of the third row.

Beispiellösung Piano



		7	5	5	6	4	2	4	Taste
Finger	1	1	2	3	3	5	5	5	
	2	1	2	3	4	5	5	5	
	3	1	2	3	4	4	5	6	
	4	1	1	2	4	4	5	6	
	5	1	1	2	3	4	5	6	

Implementierung Piano

```
void piano(vector<int> notes, int n) {  
    vector<vector<int> > eleganz(n,vector<int>(5,0));  
    vector<vector<int> > vorgaenger(n,vector<int>(5,0));  
  
    // erste Note immer möglich  
    for(int j=0;j<5;j++) eleganz[0][j]=1;
```


Implementierung Piano

```
// folgende Noten
for(int i=1;i<n;i++) {
  for(int j=0;j<5;j++) {
    // Sprung -> kein neuer Eleganzpunkt
    for(int k=0;k<5;k++) {
      if(eleganz[i-1][k]>eleganz[i][j]) {
        eleganz[i][j] = eleganz[i-1][k];
        vorgaenger[i][j] = k;
      }
    }
  }
}
```

```
// ohne Sprung: ein neuer Eleganzpunkt
int vorheriger_finger = j-(notes[i]-notes[i-1]);
if(0<=vorheriger_finger &&
   vorheriger_finger<5) {
  if(eleganz[i-1][vorheriger_finger]+1 >
     eleganz[i][j]) {
    eleganz[i][j] =
      eleganz[i-1][vorheriger_finger]+1;
    vorgaenger[i][j] = vorheriger_finger;
  } } } }
```

Implementierung Piano

```
// ermittle letzten Finger und maximale Eleganz
int sol = 0;
int last_finger;
for(int j=0;j<5;j++) {
    if(sol<eleganz[n-1][j]) {
        sol = eleganz[n-1][j];
        last_finger = j;
    }
}
cout << sol << endl;
```

Implementierung Piano

```
vector<int> belegung(n,0);
belegung[n-1] = last_finger;
for(int i=n-2;i>=0;i--) {
    belegung[i] = vorgaenger[i+1][belegung[i+1]];
}
for(int i=0;i<n;i++) {
    cout << belegung[i]+1;
    if(i<n-1) cout << " ";
    else cout << endl;
}
```


SOI-Vortrag STIU

- Vielen Dank für Ihre Aufmerksamkeit.
- Feedback Formular auf der Homepage