**DecimalToHexal**

| Line | dec | result | factor | digit | comment |
|---|---|---|---|---|---|
| 1 | 25 | | | | |
| 3 | | 0 | | | |
| 4 | | | 1 | | |
| 5 | | | | | true, dec != 0 |
| 7 | | | | 25%6=1 | |
| 8 | 25/6=4 | | | | |
| 9 | | 0+1*1=1 | | | |
| 10 | | | 1*10=10 | | |
| 5 | | | | | true, dec != 0 |
| 7 | | | | 4mod6=4 | |
| 8 | 4/6=0 | | | | |
| 9 | | 4+10*4=41 | | | |
| 10 | | | 10*10=100 | | |
| 5 | | | | | false dec=0 |
| 12 | | | | | return 41 |

| dec | result |
|---|---|
| 25 | 41 |
| 41 | 6 |
| 66 | 150 |

rechnet zahlen vom decimal(Basis 10) ins hexa (Basis 6) um

**ConvertNumberFromSystemToSystem**

| line | number | fromSystem | toSystem | result | other | system | factor | digit | dec | Kommentar |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 42 | 10 | 6 | | | | | | | |
| 3 | | | | 0 | | | | | | |
| 4 | | | | | | | | | | OtherToDecimal wird aufgerufen |
| 23 | | | | | 42 | 10 | | | | |
| 25 | | | | 0 | | | | | | |
| 26 | | | | | | | 1 | | | |
| 27 | | | | | | | | | | true, other !=0 |
| 29 | | | | | | | | 42%10=2 | | |
| 30 | | | | | 42/10=4 | | | | | |
| 31 | | | | 0+1*2=2 | | | | | | |
| 32 | | | | | | | 10 | | | |
| 27 | | | | | | | | | | true, other !=0 |
| 29 | | | | | | | | 4%10=4 | | |
| 30 | | | | | 4/10=0 | | | | | |
| 31 | | | 2*4*10=42 | | | | | | | |
| 32 | | | | | | | 10*10=100 | | | |
| 27 | | | | | | | | | | false, other=0 |
| 34 | | | | | | | | | | return 42 |
| 5 | | | | | | | | | | decimalToOther wird aufgerufen |
| 9 | | | | | | 6 | | | 42 | |
| 11 | | | | 0 | | | | | | |
| 12 | | | | | | | 1 | | | |
| 13 | | | | | | | | | | true, dec !=0 |
| 15 | | | | | | | | 42%6=0 | | |
| 16 | | | | | | | | | 42/6=7 | |
| 17 | | | | 0+1*0=0 | | | | | | |
| 18 | | | | | | | 1*10=10 | | | |
| 13 | | | | | | | | | | true, dec!=0 |
| 15 | | | | | | | | 7%6=1 | | |
| 16 | | | | | | | | | 7/6=1 | |
| 17 | | | | 0+10*1=10 | | | | | | |
| 18 | | | | | | | 10*10=100 | | | |
| 13 | | | | | | | | | | true, dec!=0 |
| 15 | | | | | | | | 1%6=1 | | |
| 16 | | | | | | | | | 1/6=0 | |
| 17 | | | | 10+*100*1=110 | | | | | | |
| 18 | | | | | | | 100*10=1000 | | | |
| 13 | | | | | | | | | | false, dec=0 |
| 20 | | | | | | | | | | return 110 |
| 6 | | | | | | | | | | return 110 |

Dieser Code wandelt Zahlen in andere zahlensysteme um. z.B binär in hexa oder hexa in dec

Der Code braucht 3 eingaben vom Benutzer. Die Zahl, die umgerechnet werden soll. Aus welchem Zahlensystem die Zahl stammt und in welches Zahlensystem es umegerechnet werden soll

Vorgehen: die Zahl wird ins Dezimalsystem umgerechent ( OtherToDecimal) und danach in das gewünschte System (decimalToOther). Hier wird nicht geprüft ob die Zahl schon im dezimal ist.

| Zahl | fromSystem | toSystem | result |
|---|---|---|---|
| 42 | 10 | 6 | 110 |
| 40 | 10 | 2 | 101000 |
| 26 | 6 | 10 | 18 |

**Optimierung**

Es wird als erstes in das diezmalsystem umgerechnet und dann im ein anderes.

Der Erste Schritt passiert auch wenn die eingegebene zahl bereits im dezimalsystem ist.

Hier könnte eine If-Anweisung dies lösen, welche überprüft ob die zahl
schon im dezimal system ist. (mit der eingeben fromSystem variable).
und gegebenfalls dann direkt die OtherToDecimal funktion ausführen