moneychanger.py

Sally Poon A01232177

Given a total balance, create a list of the currency/change needed to pay the amount in the smallest amount of change possible.

MONEYCHANGER(total_dollars)

Start → accept a total_dollars amount to be changed → define a list of canadian_currency amounts = [100, 50, 20, 10, 5, 2, 1, 0.25, 0.10, 0.05] → define an output list as empty (for now!) → send total_dollars to **ROUNDOFF_PENNIES**

assign result of **ROUNDOFF_PENNIES** to total_dollars → work with change_value in the list of canadian_currency amounts from left to right → take the *n*-th change_value → add the quotient of total_dollars divided by change_value to output → total_dollars = total_dollar - amount appended to ouput → Are we at the end of canadian_currency list?

Are we at the end of canadian_currency list?
- **Yes** → RETURN output list
- **No** → proceed to next change_value in the canadian_currency list → (back to take the *n*-th change_value)

ROUNDOFF_PENNIES(total_dollars) → assign hundredths decimal value as the last number in total_dollars → Is hundredths "."?

adjust for Python's auto-truncate of any trailing 0s

Is hundredths "."?
- **Yes** → RETURN total_dollars
- **No** ↓

Is hundredths (1, 3)?
- **Yes** → RETURN total_dollars rounded down to 1 decimal place
- **No** ↓

Is hundredths (3, 8)?
- **Yes** → RETURN total_dollars rounded up to closest 5 cents
- **No** → RETURN total_dollars rounded up to closest 10 cents