Bahria University,

Karachi Campus

COURSE: Computer Architecture and Logic Design
TERM: FALL 2024, CLASS: BSE- 3 (A)

PROJECT NAME

PROJECT TITLE:
Currency Converter

GROUP MEMBERS LIST:

Name	Enrollment
Sadaf Ashfaq	02-131232-055
Fatima Ilyas	02-131232-089
Urooj Ilyas	02-131232-107

Engr. Ramsha Mashood / Engr. Ramsha Mashood

Signed **Remarks:** **Score:**

HEADINGS:

INTRODUCTION & PROBLEM

TECHNOLOGY

FUNCTIONALITIES

MODULE DISTRIBUTION

CODE

INTERFACES

CONCLUSION

INTRODUCTION & PROBLEM:

The "Currency Converter" project is designed to address the need for quick and accurate currency conversions. Users can select currencies, input amounts, and view conversions based on predefined exchange rates. Developed in MIPS assembly language, the project provides a hands-on experience with assembly-level programming concepts such as loops, conditionals, and memory handling. The primary challenge addressed by this project is simplifying the currency conversion process for users in a console-based environment.

TECHNOLOGY :

The project is implemented in **MIPS assembly language**, focusing on:

- Arithmetic and floating-point operations for precise calculations.
- Memory management to store and retrieve conversion rates.
- Control flow to manage user interactions and program operations.
- Console-based user interface for simplicity and accessibility.

FUNCTIONALITIES:

Currency Selection

a. Users select source and target currencies from a predefined list (e.g., USD, GBP, JPY).

Amount Input and Conversion

b. Users input an amount, and the program calculates the converted value.

Exchange Rates

c. Hardcoded rates enable conversions between 45 currencies.

Error Handling

d. Ensures valid inputs for currency and amount.

Multiple Conversion Requests

e. Supports repeated conversions in a single session.

Final Output

f. Displays conversion results and exits with a thank-you message.

MODULE DISTRIBUTION

- **Fatima Ilyas:** Currency Selection and Validation, Control Flow.
- **Sadaf Ashfaq:** User Interface, Input Validation and Error Handling.
- **Urooj Ilyas:** Currency Conversion Logic, Testing and Integration.

CODE:

```
.data

main_prompt: .asciiz "-----CURRENCY CONVERTER-----\n"

name_prompt: .asciiz "Please enter your name: "

nationality_prompt: .asciiz "Please enter your nationality: "

src_prompt: .asciiz "\nChoose the currency you have:\n1: Pound (GBP) \n2: Dollar (USD) \n3: Dinar (KWD) \n4: PKR \n5: Riyal (SAR) \n6: Euro (EUR) \n7: Yen (JPY) \n8: Australian Dollar (AUD) \n9: Swiss Franc (CHF) \n10: Canadian Dollar (CAD) \n11: Indian Rupee (INR) \n12: Chinese Yuan (CNY) \n13: Brazilian Real (BRL) \n14: Russian Ruble (RUB) \n15: South Korean Won (KRW) \n16: Mexican Peso (MXN) \n17: South African Rand (ZAR) \n18: New Zealand Dollar (NZD) \n19: Singapore Dollar (SGD) \n20: Malaysian Ringgit (MYR) \n21: Norwegian Krone (NOK) \n22: Swedish Krona (SEK) \n23: Thai Baht (THB) \n24: Turkish Lira (TRY) \n25: Philippine Peso (PHP) \n26: Indonesian Rupiah (IDR) \n27: Hong Kong Dollar (HKD) \n28: UAE Dirham (AED)\n29: Saudi Riyal (SAR) \n30: Libyan DINAR(LYD)\n31: Qatari Riyal (QAR) \n32: Bahraini Dinar (BHD) \n33: Omani Rial (OMR) \n34: Bulgarian lv(BGN) \n35: Hungarian Forint(HUF) \n36: Czech koruna(CZK) \n37: Bruneian Dollar(BND) \n38: Argentine Peso (ARS) \n39: Chilean Peso (CLP) \n40: Colombian Peso (COP) \n41:
```

```
Chinese Yuan(CNY)\n42: Venezuelan Bolivar (VES) \n43: Iranian Rial (IRR) \n44: Taiwan New Dollar(TWD) \n45: Srilankan rupee(LKR) \n"

dest_prompt: .asciiz "Choose the currency to convert to:  \n[same as above]\n "

amt_prompt: .asciiz "Enter the amount to convert: "

result_msg: .asciiz "Converted amount: "

invalid_msg: .asciiz "Invalid selection. Please choose a currency between 1 and 45.\n"

continue_prompt: .asciiz "\nDo you want to perform another conversion? (1: Yes, 0: No): "

end_prompt: .asciiz "Thank you for using the Currency Converter! Have a great day!\n"

user_details: .asciiz "-----Summary-----\nUser Name: "

nationality_details: .asciiz "Nationality: "


# Conversion rates (Row=From, Column=To)

conversion_rates: .float 1, 1.255, 0.387, 349.3, 4.71, 1.207, 197.3, 2.014, 1.13, 1.808, 106.96,
        9.164, 7.78, 125.45, 1831.25, 25.33, 23.46, 2.222, 1.706, 5.639, 14.28, 13.95, 42.93,
        44.32, 73.57, 20336.87, 9.761, 4.615, 4.713, 6.161, 4.575, 0.4726, 0.4835, 2.362,
        496.54, 30.39, 1.688, 1290.94, 1243.9, 5505.64, 9.102, 6482308.86,
        53103.07, 40.81, 365.34

.float 0.798, 1, 0.308, 277.97, 3.75, 0.9597, 157.91, 1.607, 0.8989, 1.441,
        85.21, 7.298, 6.742, 99.99, 1468.76, 20.22, 18.87, 1.778, 1.358, 4.469, 11.39, 11.04, 34.12,
        35.16, 58, 16194.72, 7.767, 3.672, 3.75, 4.901, 3.64, 0.376, 0.3849, 1.876,
        394.38, 24.11, 1.358, 1023.64, 989.14, 4403.46, 7.298, 5157377,
        42.19176, 32.77, 293.48

.float 2.589, 3.244, 1, 901.89, 12.16, 3.113, 512.47, 5.215, 2.196, 4.677, 276.49,
        23.67, 21.87, 324.42, 474.64, 65.58, 61.2, 5.771, 4.408, 14.49, 36.96, 35.84, 11.74,
        114.09, 188.28, 52533.33, 25.2, 11.91, 12.16, 15.89, 11.8, 1.219, 1.248, 6.089,
        1279.62, 78.24, 4.408, 3321.53, 3209, 14310.06, 23.67, 16733227, 136898.17,
        106.31, 951.84

.float 0.00287, 0.003592, 0.001109, 1, 0.01349, 0.003452, 0.5681, 0.00578,
        0.003233, 0.005185, 0.3068, 0.02625, 0.02424, 0.3596, 5.281,
        0.07272, 0.06785, 0.006397, 0.00488, 0.01607, 0.04098, 0.03974,
        0.1228, 0.1265, 0.2088, 58.26, 0.027, 0.0132, 0.0134, 0.0176,
        0.01308, 0.001352, 0.001384, 0.006749, 1.418, 0.08671, 0.004887, 3.679,
        3.556, 15.01, 0.02619, 18543.22, 151.31, 0.1178, 1.055
```

```
.float 0.213, 0.2666, 0.08222, 74.28, 1, 0.2576, 41.96, 0.4311, 0.242,
    0.3838, 22.82, 1.946, 1.649, 30.03, 349.11, 5.568, 5.041, 0.4768,
    0.3645,
    1.192, 3.3036, 2.953, 9.162, 9.432, 15.51, 4336.7, 2.071, 0.9793,
    1, 1.312,
    0.9706, 0.1002, 0.1026, 0.504, 106.02, 6.491, 0.3645, 274.93,
    265.15, 1175.22, 1.946, 1385680.51, 11223.53, 8.737, 78.18

.float 0.827, 1.035, 0.3191, 288.43, 3.882, 1, 162.88, 1.673, 0.9395, 1.489,
    88.6, 7.557, 6.402, 116.46, 1530.56, 21.59, 19.56, 1.85, 1.415, 4.628, 11.78, 11.45,
    35.55, 36.61, 60.18, 16838.36, 8.042, 3.802, 3.882, 5.094, 3.768, 0.3892, 0.3986,
    1.955, 411.5, 25.19, 1.415, 1067.43, 1029.48, 4562.29, 7.557, 5380102.91,
    43558.27, 33.92, 303.45

.float 0.005083, 0.006359, 0.001961, 1.771, 0.0238, 0.006141, 1, 0.1027,
    0.00577, 0.009147, 0.5441, 0.04641, 0.03932, 0.7156, 9.4,
    0.1326, 0.1201, 0.01136, 0.008687, 0.02842, 0.07238, 0.07035,
    0.2183, 0.24448, 0.37, 103.45, 0.0494, 0.02335, 0.02384,
    0.03128, 0.02314, 0.002391, 0.002446, 0.01201, 2.526, 0.1546,
    0.008688, 6.555, 6.322, 28.02, 0.04641, 33043.22, 267.45, 0.2083, 1.864

.float 4946, 0.619, 0.1908, 172.35, 2.321, 0.5976, 97.39, 1, 0.5618,
    0.8901, 52.98, 4.519, 3.828, 69.32, 915.05, 12.92, 11.69, 1.106, 0.8458, 2.768,
    7.046, 6.849, 21.26, 21.89, 35.96, 10065.54, 4.81, 2.274, 2.322, 3.046, 2.253, 0.2328,
    0.2384, 1.168, 245.95, 15.05, 0.8457, 638.39, 615.76, 2729.3,
    4.519, 3217264.26, 26056.17, 20.28, 181.51

.float 0.8804, 1.102, 0.3397, 306.81, 4.132, 1.063, 173.32, 1.78, 1, 1.584, 94.29,
    8.043, 6.814, 124.01, 1628.83, 23, 20.81, 1.969, 1.505, 4.927, 12.54, 12.18, 37.82,
    38.95, 64, 17929.32, 8.559, 4.046, 4.132, 5.421, 4.01, 0.4143, 0.4242, 2.08,
    437.84, 26.79, 1.505, 1135.63, 1095.51, 4854.71, 8.042, 5724226.16,
    46354.47, 36.09, 322.92

.float 0.5554, 0.6952, 0.2143, 193.55, 2.607, 0.6713, 109.35, 1.123, 0.631,
    1, 54.49, 5.074, 4.299, 78.03, 1027.73, 14.52, 13.13, 1.242, 0.95, 3.108, 7.917, 7.692,
    23.87, 24.58, 40.38, 11313.81, 5.401, 2.553, 2.607, 3.421, 2.531, 0.2614, 0.2677,
    1.313, 276.3, 16.91, 0.9501, 716.61, 691.29, 3063.63, 5.075, 3612275.39,
    29228.64, 22.77, 203.73

.float 0.009341, 0.01168, 0.003603, 3.252, 0.04382, 0.01128, 1.837, 0.01888,
    0.0106, 0.0168, 1, 0.08529, 0.07226, 1.307, 17.27, 0.2442,
    0.2207, 0.02088, 0.01596, 0.05225, 0.133, 0.1293, 0.4012,
    0.4131, 0.6786, 190.1, 0.09077, 0.04291, 0.04382, 0.05749,
    0.04253, 0.004394, 0.004499, 0.02207, 4.643, 0.2843, 0.01596, 12.04,
    11.61, 51.5, 0.08529, 60720.82, 490.93, 0.3828, 3.424
```

```
.float 0.1095, 0.137, 0.04224, 38.13, 0.5137, 0.1322, 21.55, 0.2212, 0.1243,
       0.1969, 11.72, 1, 0.8472, 15.34, 202.47, 2.86, 2.586, 0.2447,
       0.1871, 0.6126, 1.559, 1.515, 4.704, 4.843, 7.957, 2222.63, 1.064, 0.5031,
       0.5137, 0.674, 0.4986, 0.05151, 0.05274, 0.2587, 54.43, 3.333,
       0.1871, 141.22, 136.22, 603.75, 1, 711856.99, 5763.43, 4.492,
       40.16

.float 0.1291, 0.1616, 0.04985, 45.01, 0.6063, 0.156, 25.44, 0.2611, 0.1467,
       0.2324, 13.83, 1.18, 1, 18.18, 239.01, 3.376, 3.052, 0.2889, 0.2208,
       0.723, 1.84, 1.788, 5.551, 5.716, 9.391, 2630.86, 1.255, 0.5938, 0.6063,
       0.7956, 0.5885, 0.06079, 0.06225, 0.3052, 64.23, 3.933, 0.2208,
       166.7, 160.79, 712.69, 1.18, 840228.06, 6802.25, 5.302, 47.4

.float 0.007162, 0.009038, 0.002787, 2.487, 0.03351, 0.008624, 1.411, 0.01448,
       0.008141, 0.01289, 0.7575, 0.06503, 0.0551, 1, 13.16, 0.1858,
       0.1681, 0.01598, 0.01221, 0.04001, 0.1018, 0.9929, 0.3082,
       0.3174, 0.5215, 145.95, 0.06922, 0.03273, 0.03342, 0.04385,
       0.03244, 0.003341, 0.003421, 0.01677, 3.547, 0.2171, 0.01219, 9.205,
       8.879, 39.19, 0.06491, 46210.9, 373.65, 0.2916, 2.619

.float 0.00054, 0.000676, 0.000209, 0.1883, 0.002537, 0.000653, 0.1064,
       0.001093, 0.000614, 0.000973, 0.05789, 0.000924, 0.004184, 0.07541,
       1, 0.01411, 0.01277, 0.001209, 0.000924, 0.003025, 0.007701,
       0.007486, 0.02323, 0.02391, 0.0393, 10.97, 0.005256, 0.002485,
       0.002538, 0.003329, 0.002463, 0.000254, 0.00026, 0.001277, 0.2688,
       0.01645, 0.000924, 0.6974, 0.6727, 2.981, 0.00493, 3515.29, 28.44,
       0.02221, 0.1983

.float 0.03829, 0.04794, 0.01478, 13.34, 0.1798, 0.04628, 7.544, 0.07745,
       0.04351, 0.06895, 4.104, 0.3501, 0.2966, 5.371, 70.9, 1, 0.9053,
       0.0857, 0.0655, 0.2144, 0.546, 0.5305, 1.647, 1.696, 2.786, 780.49,
       0.3725, 0.1761, 0.1798, 0.236, 0.1746, 0.01803, 0.01847,
       0.09056, 19.05, 1.166, 0.06549, 49.44, 47.69, 211.33, 0.35, 249181.89,
       2016.89, 1.573, 14.05

.float 0.04231, 0.05297, 0.01633, 14.74, 0.1986, 0.05114, 8.335, 0.08559,
       0.040807, 0.07616, 4.533, 0.3867, 0.3276, 5.928, 78.29, 1.104, 1,
       0.09467, 0.07235, 0.2368, 0.6029, 0.5861, 1.818, 1.873, 3.077,
       861.99, 0.4115, 0.1945, 0.1986, 0.2606, 0.1928, 0.01992,
       0.02039, 0.1, 21.04, 1.288, 0.07234, 54.6, 52.67, 233.46, 0.3866,
       275265.76, 2224.38, 1.738, 15.53

.float 0.4469, 0.5595, 0.1725, 155.78, 2.098, 0.5402, 88.01, 0.904, 0.5076,
       0.8045, 47.88, 1.056, 3.46, 62.66, 827.01, 11.66, 10.56, 1, 0.7641, 2.502,
       6.368, 6.191, 19.2, 19.79, 32.5, 9095.61, 4.347, 2.055, 2.098, 2.753, 2.037, 0.2104,
```

0.2154, 1.056, 222.35, 13.61, 0.7642, 576.91, 556.47, 2466.46, 4.084, 2901811.36, 23475.22, 18.34, 164.05

.float 0.5851, 0.7323, 0.2257, 203.83, 2.745, 0.7072, 115.16, 1.183, 0.6648, 1.053, 62.76, 5.345, 4.529, 83.3, 1082.13, 15.27, 13.83, 1.308, 1, 3.273, 8.335, 8.103, 25.14, 25.89, 42.53, 11894.9, 5.689, 2.689, 2.745, 3.602, 2.665, 0.2752, 0.2818, 1.383, 291.13, 17.82, 1, 755.01, 728.09, 3227.1, 5.344, 3804654.06, 30813.28, 23.99, 214.69

.float 0.1787, 0.2236, 0.06897, 62.26, 0.8385, 0.216, 35.17, 0.3613, 0.203, 0.3217, 19.15, 1.632, 1.383, 25.35, 330.45, 4.665, 4.224, 0.3996, 0.3053, 1, 2.546, 2.475, 7.678, 7.906, 12.99, 3637, 1.737, 0.8213, 0.8386, 1.1, 0.814, 0.08408, 0.08609, 0.4224, 88.9, 5.443, 0.3053, 230.55, 222.38, 985.68, 1.632, 1162074.37, 9411.69, 7.325, 65.55

.float 0.07018, 0.08783, 0.02707, 24.45, 0.3293, 0.08482, 13.81, 0.1419, 0.07973, 0.1263, 7.525, 0.6411, 0.5432, 9.925, 129.74, 1.831, 1.658, 0.1569, 0.1199, 0.3927, 1, 0.9721, 3.015, 3.104, 5.101, 1427.75, 0.6823, 0.3225, 0.3293, 0.4321, 0.3197, 0.03302, 0.03381, 0.1659, 34.91, 2.137, 0.1199, 90.54, 87.33, 387.11, 0.6411, 456395.97, 3694.9, 2.876, 25.75

.float 0.9721, 0.08783, 0.02708, 24.45, 0.3293, 0.08482, 13.81, 0.1419, 0.07973, 0.1263, 7.523, 0.6411, 0.5431, 9.989, 129.82, 1.832, 1.659, 0.1569, 0.1199, 0.3927, 1, 1, 3.014, 3.101, 5.101, 1425.2, 0.6823, 0.3225, 0.3293, 0.432, 0.3197, 0.03302, 0.03381, 0.1659, 34.91, 2.137, 0.1199, 90.54, 87.33, 387.08, 0.6411, 456369.98, 3683.68, 2.877, 25.74

.float 0.02329, 0.02915, 0.008988, 8.115, 0.1093, 0.02815, 4.585, 0.04711, 0.02645, 0.04191, 2.496, 0.2126, 0.1802, 3.312, 43.05, 0.6079, 0.5502, 0.05207, 0.03978, 0.1302, 0.3316, 0.3224, 1, 1.03, 1.691, 473.88, 0.2263, 0.1069, 0.1092, 0.1433, 0.106, 0.01095, 0.01121, 0.05501, 11.57, 0.7088, 0.03977, 30.02, 28.96, 128.35, 0.2125, 151331.21, 1225.53, 0.9539, 8.538

.float 0.02264, 0.02833, 0.008742, 7.881, 0.1061, 0.02734, 4.452, 0.04574, 0.02569, 0.04072, 2.425, 0.2066, 0.175, 3.199, 41.81, 0.5904, 0.5344, 0.05056, 0.03864, 0.1265, 0.322, 0.313, 0.9712, 1, 1.642, 460.03, 0.2197, 0.1038, 0.106, 0.1391, 0.1029, 0.01063, 0.01088, 0.05342, 11.24, 0.6883, 0.03862, 29.15, 28.12, 124.64, 0.2064, 146961.09, 1190.24, 0.9263, 8.293

.float 0.01375, 0.01721, 0.005308, 4.792, 0.06456, 0.01661, 2.708, 0.02781, 0.01562, 0.02475, 1.473, 0.1256, 0.1064, 1.918, 25.44, 0.3587,

0.3249, 0.03076, 0.02351, 0.07698, 0.1959, 0.1904, 0.5909,
0.6089, 1, 279.57, 0.1337, 0.06322, 0.06456, 0.0847,
0.06266, 0.006473, 0.006628, 0.0325, 6.84, 0.4187, 0.02351, 17.74,
17.11, 75.88, 0.1256, 89458.8, 722.26, 0.5648, 5.047

.float 0.000049, 0.000061, 0.000019, 0.0171, 0.00023, 0.000059, 0.0096,
0.000099, 0.000056, 0.000088, 0.0052, 0.00045, 0.00038, 0.0069,
0.09097, 0.0012, 0.0011, 0.00011, 0.000084, 0.000275, 0.000701,
0.00068, 0.00211, 0.002174, 0.00357, 1, 0.000478, 0.000226,
0.000231, 0.000302, 0.000224, 0.000023, 0.000024, 0.000116, 0.02442,
0.0015, 0.000084, 0.06353, 0.06128, 0.2716, 0.00045, 320.23,
2.578, 0.002019, 0.01803

.float 0.1028, 0.1287, 0.03969, 35.84, 0.4828, 0.1242, 20.26, 0.2079,
0.1168, 0.1851, 11.01, 0.9397, 0.7962, 14.4, 190.27, 2.685, 2.43,
0.23, 0.1758, 0.5756, 1.465, 1.424, 4.42, 4.545, 7.478, 2093.38, 1,
0.4728, 4828, 0.6334, 0.4686, 0.04841, 0.04957, 0.243, 51.15, 3.131,
0.1758, 132.73, 128.02, 567.5, 0.9398, 669062.25, 5409.71, 4.223,
37.74

.float 0.2174, 0.2722, 0.0839, 75.79, 1.021, 0.2627, 42.85, 0.4396, 0.2471,
0.3914, 23.3, 1.987, 1.683, 30.35, 402.4, 5.679, 5.139, 0.4864, 0.3718, 1.217,
3.098, 3.01, 9.348, 9.626, 15.81, 4426.16, 2.114, 1, 1.021, 1.339, 0.9911,
0.1023, 0.1048, 0.5139, 108.16, 6.622, 0.3718, 280.71,
270.76, 1200.1, 1.987, 1414908.67, 11422.53, 8.936, 79.82

.float 0.2129, 0.2666, 0.08222, 74.23, 1, 0.2574, 41.96, 0.4307, 0.242,
0.3834, 22.82, 1.946, 1.649, 29.79, 394.14, 5.568, 5.035, 0.4766, 0.3642,
1.192, 3.036, 2.949, 9.156, 9.427, 15.48, 4337.52, 2.071, 0.9793, 1, 1.312,
0.9706, 0.1002, 0.1026, 0.5034, 105.94, 6.486, 0.3642,
274.91, 265.17, 1175.3, 1.946, 1385610.94, 11184.63, 8.737, 78.18

.float 0.1624, 0.2032, 0.06266, 56.57, 0.7621, 0.1962, 31.96, 0.3283,
0.1844, 0.2922, 17.39, 1.483, 1.256, 22.85, 300.41, 4.245, 3.839, 0.3633,
0.2777, 0.909, 2.315, 2.249, 6.98, 7.188, 11.8, 3300.87, 1.579, 0.7464,
0.7622, 1, 0.7398, 0.07642, 0.07825, 0.3839, 80.77, 4.845,
0.2777, 209.52, 202.1, 895.76, 1.483, 1055307.01, 8524.54, 6.659, 59.58

.float 0.2195, 0.2747, 0.0847, 76.47, 1.03, 0.2652, 43.2, 0.4439, 0.2493,
0.3951, 23.51, 2.005, 1.699, 30.74, 406.08, 5.741, 5.191, 0.491, 0.3754, 1.228,
3.128, 3.039, 9.433, 9.713, 15.95, 4457.17, 2.134, 1.008, 1.03, 1.351, 1, 0.1032,
0.1057, 0.5188, 109.17, 6.683, 0.3754, 283.21, 273.17,
1210.82, 2.005, 1427544.67, 11551.9, 9.001, 80.54

```
.float 2.124,  2.659,  0.8199,  740.36,  9.973,  2.566,  418.24,  4.295,  2.412,  3.824,
       227.59,  19.41,  16.44,  298.5,  3931.45,  55.52,  50.23,  4.752,  3.633,  11.89,  30.28,  29.41,
       91.3,  94.04,  154.47,  43106.16,  20.66,  9.767,  9.973,  13.08,  9.68,  1,  1.023,  5.021,
       1056.73,  64.67,  3.633,  2741.69,  2644.67,  11721.89,  19.41,  13819576.68,
       111855.13,  87.13,  779.6

.float 2.076,  2.597,  0.8008,  723.72,  9.739,  2.508,  408.45,  4.196,  2.356,  3.736,
       222.26,  18.95,  16.06,  291.81,  3839.99,  54.2,  49.07,  4.641,  3.548,  11.61,  29.56,
       28.73,  89.16,  91.79,  151.14,  42259.89,  20.17,  9.538,  9.739,  12.77,  9.454,  0.9765,
       1,  4.905,  1032.22,  63.19,  3.548,  2677.43,  2582.51,  11445.24,  18.95,
       13496388.32,  109301.32,  85.08,  761.35

.float 0.4228,  0.5292,  0.1631,  147.47,  1.984,  0.5112,  83.28,  0.8556,
       0.4803,  0.7616,  42.29,  3.863,  3.273,  59.38,  782.46,  11.04,  10,  0.9462,
       0.7234,  2.366,  6.027,  5.858,  18.18,  18.72,  30.77,  8607.7,  4.111,  1.943,  1.985,  2.604,
       1.926,  0.199,  0.2038,  1,  210.39,  12.88,  0.7234,  545.74,  526.36,
       2332.63,  3.863,  2750697.62,  22266.41,  17.34,  155.14

.float 0.002,  0.0025,  0.0007,  0.7004,  0.0094,  0.0024,  0.3958,
       0.00406,  0.0022,  0.0036,  0.2151,  0.01835,  0.0155,  0.2844,
       3.718,  0.05245,  0.04754,  0.0044,  0.0034,  0.01124,  0.02863,
       0.02785,  0.08641,  0.089,  0.1464,  40.88,  0.01954,  0.0092,  0.0094,
       0.01237,  0.0091,  0.00094,  0.00096,  0.0047,  1,  0.06122,
       0.0034,  2.593,  2.5,  11.08,  0.01835,  13068.71,  105.82,  0.08241,  0.7374

.float 0.03283,0.04108,  0.01266,  11.44,  0.154,  0.03969,  6.466,  0.06645,  0.03729,
       0.05916,  3.514,  0.2998,  0.254,  4.646,  60.72,  0.8566,  0.7765,  0.07348,
       0.05617,  0.1836,  0.4676,  0.4548,  1.41,  1.452,  2.39,  666.75,  0.319,
       0.1508,  0.154,  0.2021,  0.1495,  0.01544,  0.01581,  0.0776,  16.33,
       1,  0.0561,  42.35,  40.84,  181.03,  0.2998,  213457.61,  1728.72,  1.344,
       12.04

.float 0.5846,  0.7317,  0.2256,  203.9,  2.743,  0.7068,  115.2,  1.183,  0.6639,  1.053,
       62.6,  5.34,  4.525,  82.81,  1081.69,  15.24,  13.83,  1.308,  1,  3.27,  8.321,  8.095,  25.11,
       25.87,  42.48,  11870.39,  5.681,  2.686,  2.742,  3.598,  2.662,  0.02749,  0.2815,  1.381,
       290.64,  17.79,  1,  753.94,  727.04,  3221.73,  5.337,  3799706.04,
       30758.39,  23.95,  214.46

.float 0.0007,  0.0009,  0.0002,  0.2702,  0.0036,  0.0009,  0.1526,
       0.0015,  0.0008,  0.0013,  0.08301,  0.007,  0.006,  0.1097,  1.434,
       0.02023,  0.01831,  0.0017,  0.0013,  0.0043,  0.01102,  0.01073,
       0.03329,  0.03431,  0.05634,  15.73,  0.0075,  0.0035,  0.0036,
       0.0047,  0.0035,  0.00036,  0.00037,  0.0018,  0.3855,  0.02359,
       0.0013,  1,  0.9644,  4.273,  0.00708,  5040.01,  40.83,  0.03175,  0.2844
```

```
.float 0.0008, 0.001, 0.0003, 0.2802, 0.0037, 0.0009, 0.1576, 0.0016,
       0.0009, 0.0014, 0.086, 0.0073, 0.0062, 0.1105, 1.483, 0.0208,
       0.0189, 0.0017, 0.0013, 0.0044, 0.0114, 0.01109, 0.0345,
       0.0355, 0.0583, 16.27, 0.0078, 0.0036, 0.0037, 0.0049,
       0.0036, 0.00037, 0.00038, 0.00189, 0.3987, 0.0245, 0.0013,
       1.036, 1, 4.432, 0.0073, 5226.06, 42.32, 0.0329, 0.2948

.float 0.0001, 0.0002, 0.00007, 0.0631, 0.0008, 0.0002, 0.03557,
       0.0003, 0.0002, 0.0003, 0.0193, 0.00165, 0.0014, 0.0249,
       0.3342, 0.0047, 0.0042, 0.0004, 0.0003, 0.001, 0.00257,
       0.0025, 0.0077, 0.008, 0.0131, 3.668, 0.0017, 0.0008, 0.0008,
       0.0011, 0.0008, 0.00008, 0.00008, 0.0004, 0.0898, 0.0054,
       0.0003, 0.2337, 0.2253, 1, 0.0016, 1178.05, 9.538, 0.0074,
       0.06645

.float 0.1093, 0.137, 0.04222, 38.18, 0.5137, 0.1319, 21.5, 0.221, 0.124, 0.1972,
       11.72, 1, 0.8475, 15.05, 202.1, 2.84, 2.585, 0.2443, 0.1868, 0.6124,
       1.554, 1.51, 4.702, 4.843, 7.951, 2218.75, 1.063, 0.503, 0.513, 0.674, 0.4986,
       0.0515, 0.0527, 0.258, 54.29, 3.321, 0.1868, 141.24, 136.21,
       604.38, 1, 711880.52, 5765.17, 4.488, 40.17

.float 0, 0, 0, 0.000054, 0.000001, 0, 0.00003, 0, 0, 0,
       0.000016, 0.000001, 0.000001, 0.00021, 0.00028, 0.000004, 0.000004,
       0, 0, 0.000001, 0.000002, 0.000002, 0.000007, 0.000007, 0.000011,
       0.00311, 0.000001, 0.000001, 0.000001, 0.000001, 0.000001, 0, 0,
       0, 0.000076, 0.000005, 0, 0.00019, 0.00019, 0.00084, 0.000001,
       1, 0.0081, 0.000006, 0.00005

.float 0.000019, 0.000024, 0.000007, 0.0066, 0.000089, 0.000023, 0.0037,
       0.000038, 0.000022, 0.000034, 0.00203, 0.00017, 0.00014, 0.0026,
       0.035, 0.0004, 0.0004, 0.00004, 0.00003, 0.0001, 0.0002,
       0.00026, 0.0008, 0.00084, 0.0013, 0.3848, 0.0001, 0.00008,
       0.000089, 0.00011, 0.00008, 0.000009, 0.000009, 0.00004, 0.0094,
       0.0005, 0.00003, 0.0244, 0.0236, 0.1048, 0.0001, 123.51,
       1, 0.0007, 0.0069

.float 0.0243, 0.0304, 0.0093, 8.502, 0.1143, 0.0292, 4.78, 0.049, 0.0275,
       0.0438, 2.61, 0.2225, 0.1888, 3.285, 44.91, 0.6319, 0.5726,
       0.0542, 0.0415, 0.1363, 0.3455, 0.3356, 1.046, 1.078, 1.768,
       493.03, 0.2369, 0.112, 0.1144, 0.1504, 0.111, 0.0114, 0.117,
       0.0572, 12.03, 0.7368, 0.0415, 31.45, 30.32, 134.54, 0.2226,
       158521.92, 1284.07, 1, 8.912

.float 0.0027, 0.0034, 0.001, 0.951, 0.0128, 0.0032, 0.5393, 0.0054, 0.003,
       0.0049, 0.2917, 0.0249, 0.0211, 0.3611, 5.036, 0.0694,
```

```
        0.0638,         0.006,  0.0046,         0.0152,         0.0388,         0.0376,         0.1164,
        0.1201,         0.1977,         55.27,  0.0265,         0.0125,         0.0128,         0.0167,
        0.0124,         0.0012,         0.0013,         0.0064,         1.348,  0.0284,         0.0046,         3.52,
        3.393,  15.05,  0.0249,         17672.11,       144.43,         0.1121,         1

# Temporary storage for user input
user_name: .space 50
user_nationality: .space 50

.text
.globl main
main:
   # Print main prompt
   li $v0, 4
   la $a0, main_prompt
   syscall

   # Get user's name
   li $v0, 4
   la $a0, name_prompt
   syscall

   li $v0, 8          # Read string syscall
   la $a0, user_name
   li $a1, 50         # Maximum string length
   syscall

   # Get user's nationality
```

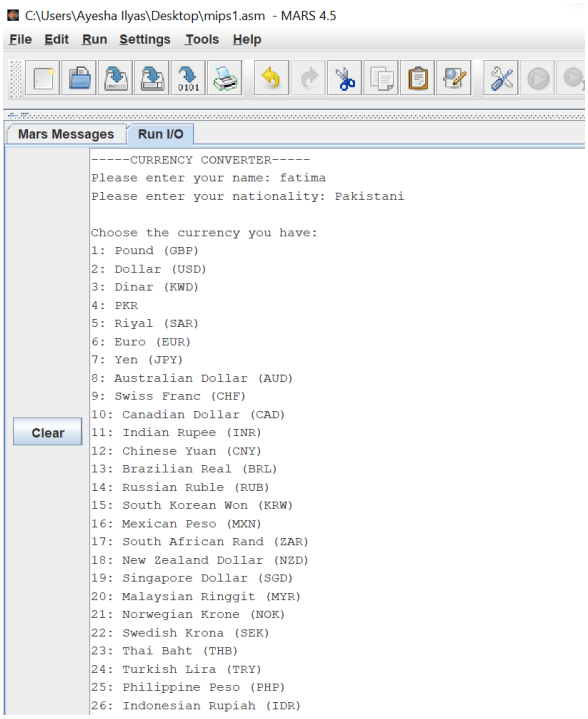
```
    li $v0, 4
    la $a0, nationality_prompt
    syscall

    li $v0, 8
    la $a0, user_nationality
    li $a1, 50
    syscall

    # Initialize loop
    li $t6, 1           # Continue flag (1 means continue)

do_while_loop:
    # Prompt for source currency
    li $v0, 4
    la $a0, src_prompt
    syscall

    li $v0, 5           # Read integer syscall
    syscall
    move $t0, $v0       # $t0 = fromCurrency

    # Prompt for destination currency
    li $v0, 4
    la $a0, dest_prompt
    syscall
```

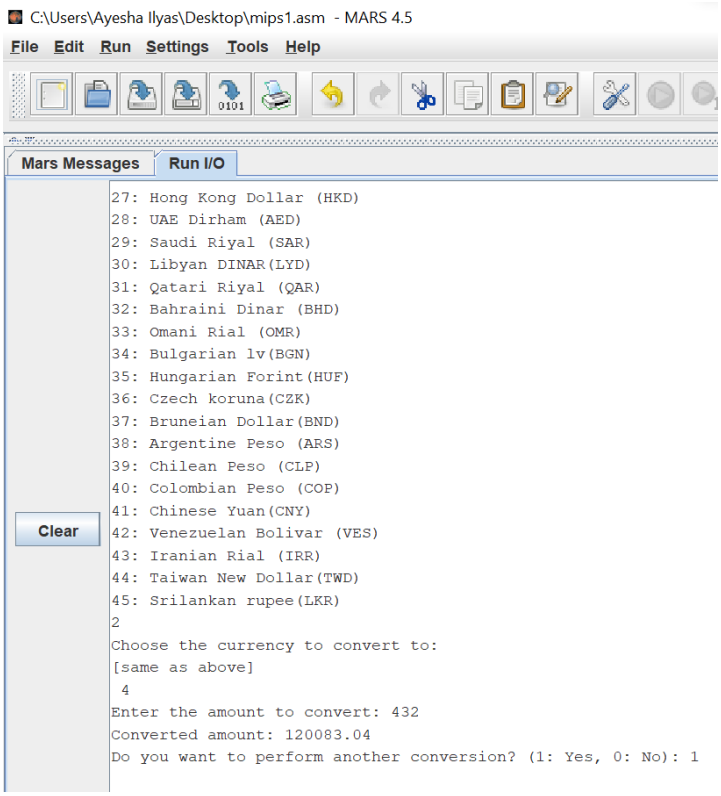
```
li $v0, 5
syscall
move $t1, $v0         # $t1 = toCurrency

# Validate inputs
blt $t0, 1, invalid_input
bgt $t0, 45, invalid_input
blt $t1, 1, invalid_input
bgt $t1, 45, invalid_input

# Prompt for amount
li $v0, 4
la $a0, amt_prompt
syscall

li $v0, 6
syscall
mov.s $f0, $f0        # $f0 = amount

# Conversion logic
la $a0, conversion_rates
sub $t2, $t0, 1
sub $t3, $t1, 1
li $t4, 45
mul $t5, $t2, $t4
add $t5, $t5, $t3
sll $t5, $t5, 2
```

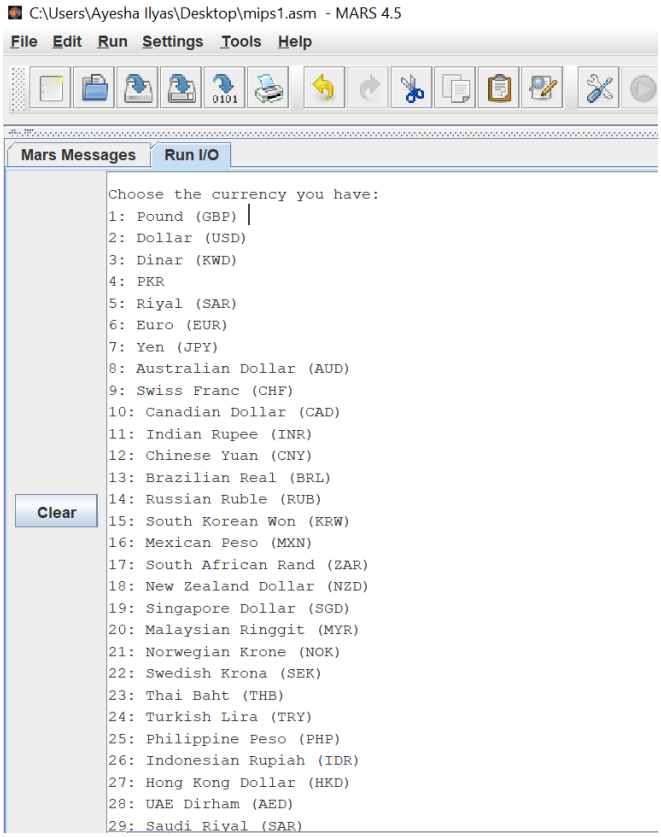
```
    add $a0, $a0, $t5
    lwc1 $f1, 0($a0)
    mul.s $f2, $f0, $f1

    # Print result
    li $v0, 4
    la $a0, result_msg
    syscall

    li $v0, 2
    mov.s $f12, $f2
    syscall

    # Ask to continue
    li $v0, 4
    la $a0, continue_prompt
    syscall

    li $v0, 5
    syscall
    move $t6, $v0
    bne $t6, 1, exit_loop

    j do_while_loop

exit_loop:
    # Print summary
```

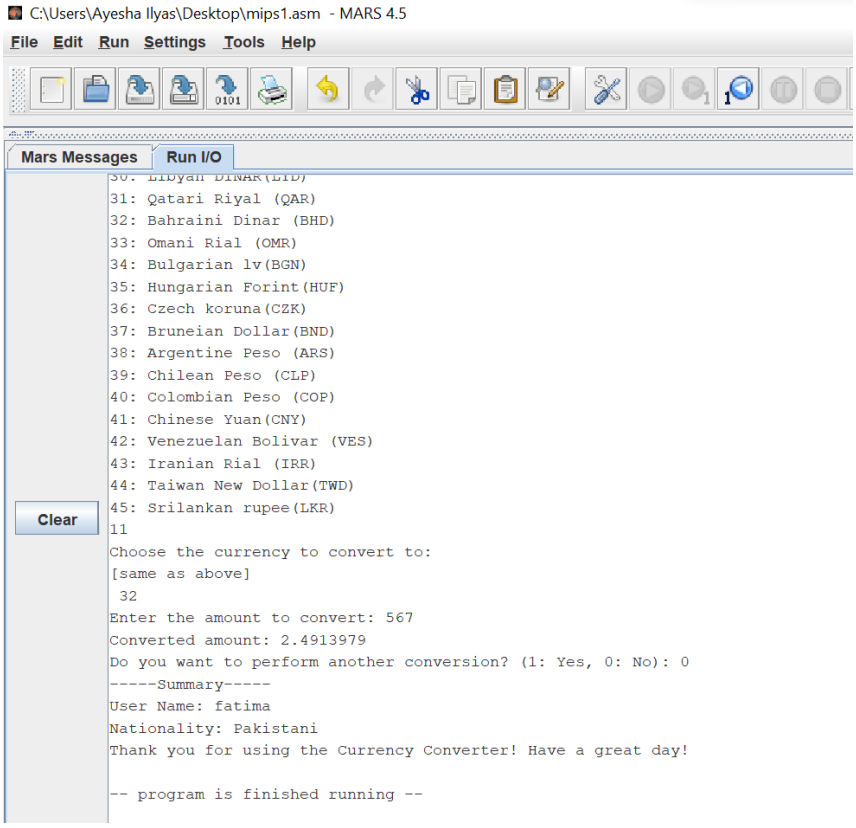
```
    li $v0, 4
    la $a0, user_details
    syscall

    la $a0, user_name
    syscall

    li $v0, 4
    la $a0, nationality_details
    syscall

    la $a0, user_nationality
    syscall

    li $v0, 4
    la $a0, end_prompt
    syscall

    li $v0, 10
    syscall

invalid_input:
    li $v0, 4
    la $a0, invalid_msg
    syscall

    j do_while_loop
```

INTERFACES:


C:\Users\Ayesha Ilyas\Desktop\mips1.asm - MARS 4.5
File Edit Run Settings Tools Help
Mars Messages
Run I/O
Clear
-----CURRENCY CONVERTER-----
Please enter your name: fatima
Please enter your nationality: Pakistani
Choose the currency you have:
1: Pound (GBP)
2: Dollar (USD)
3: Dinar (KWD)
4: PKR
5: Riyal (SAR)
6: Euro (EUR)
7: Yen (JPY)
8: Australian Dollar (AUD)
9: Swiss Franc (CHF)
10: Canadian Dollar (CAD)
11: Indian Rupee (INR)
12: Chinese Yuan (CNY)
13: Brazilian Real (BRL)
14: Russian Ruble (RUB)
15: South Korean Won (KRW)
16: Mexican Peso (MXN)
17: South African Rand (ZAR)
18: New Zealand Dollar (NZD)
19: Singapore Dollar (SGD)
20: Malaysian Ringgit (MYR)
21: Norwegian Krone (NOK)
22: Swedish Krona (SEK)
23: Thai Baht (THB)
24: Turkish Lira (TRY)
25: Philippine Peso (PHP)
26: Indonesian Rupiah (IDR)


C:\Users\Ayesha Ilyas\Desktop\mips1.asm - MARS 4.5
File Edit Run Settings Tools Help
Mars Messages
Run I/O
Clear
27: Hong Kong Dollar (HKD)
28: UAE Dirham (AED)
29: Saudi Riyal (SAR)
30: Libyan DINAR(LYD)
31: Qatari Riyal (QAR)
32: Bahraini Dinar (BHD)
33: Omani Rial (OMR)
34: Bulgarian lv(BGN)
35: Hungarian Forint(HUF)
36: Czech koruna(CZK)
37: Bruneian Dollar(BND)
38: Argentine Peso (ARS)
39: Chilean Peso (CLP)
40: Colombian Peso (COP)
41: Chinese Yuan(CNY)
42: Venezuelan Bolivar (VES)
43: Iranian Rial (IRR)
44: Taiwan New Dollar(TWD)
45: Srilankan rupee(LKR)
2
Choose the currency to convert to:
[same as above]
4
Enter the amount to convert: 432
Converted amount: 120083.04
Do you want to perform another conversion? (1: Yes, 0: No): 1


C:\Users\Ayesha Ilyas\Desktop\mips1.asm - MARS 4.5
File Edit Run Settings Tools Help
Mars Messages
Run I/O
Clear
Choose the currency you have:
1: Pound (GBP)
2: Dollar (USD)
3: Dinar (KWD)
4: PKR
5: Riyal (SAR)
6: Euro (EUR)
7: Yen (JPY)
8: Australian Dollar (AUD)
9: Swiss Franc (CHF)
10: Canadian Dollar (CAD)
11: Indian Rupee (INR)
12: Chinese Yuan (CNY)
13: Brazilian Real (BRL)
14: Russian Ruble (RUB)
15: South Korean Won (KRW)
16: Mexican Peso (MXN)
17: South African Rand (ZAR)
18: New Zealand Dollar (NZD)
19: Singapore Dollar (SGD)
20: Malaysian Ringgit (MYR)
21: Norwegian Krone (NOK)
22: Swedish Krona (SEK)
23: Thai Baht (THB)
24: Turkish Lira (TRY)
25: Philippine Peso (PHP)
26: Indonesian Rupiah (IDR)
27: Hong Kong Dollar (HKD)
28: UAE Dirham (AED)
29: Saudi Riyal (SAR)


C:\Users\Ayesha Ilyas\Desktop\mips1.asm - MARS 4.5
File Edit Run Settings Tools Help
Mars Messages
Run I/O
Clear
30: Libyan DINAR(LYD)
31: Qatari Riyal (QAR)
32: Bahraini Dinar (BHD)
33: Omani Rial (OMR)
34: Bulgarian lv(BGN)
35: Hungarian Forint(HUF)
36: Czech koruna(CZK)
37: Bruneian Dollar(BND)
38: Argentine Peso (ARS)
39: Chilean Peso (CLP)
40: Colombian Peso (COP)
41: Chinese Yuan(CNY)
42: Venezuelan Bolivar (VES)
43: Iranian Rial (IRR)
44: Taiwan New Dollar(TWD)
45: Srilankan rupee(LKR)
11
Choose the currency to convert to:
[same as above]
32
Enter the amount to convert: 567
Converted amount: 2.4913979
Do you want to perform another conversion? (1: Yes, 0: No): 0
-----Summary-----
User Name: fatima
Nationality: Pakistani
Thank you for using the Currency Converter! Have a great day!
-- program is finished running --

CONCLUSION:

The Currency Converter project demonstrates the practical application of MIPS assembly language for solving real-world problems. By integrating fundamental programming concepts such as loops, conditionals, and error handling, the project achieves its goal of a functional and efficient currency conversion tool. This project serves as a valuable educational experience for mastering low-level programming techniques while addressing user needs.