RMR = 10\*weight + 6.25\*height - 5\*age + s

TDEE = RMR \* activity level

Calories needed for losing weight = tdee \*0.8,

Calories needed for maintaining weight = tdee,

Calories needed for gaining weight = tdee \*1.2,

Recommended goal = tdee \* 0.2,

Maximum goal calories = rmr \*2.5,

Speed = x / y

calories burned in one minute, c1 = z / y

calories yet to burn in order to achieve the daily goal, c2 = goal - z

time needed to achieve goal = c2 / c1

END

START

Program(String program)

Price1(String price1)

TSHouse(String tsHouse)

main(String[] args)

Output mini game, discount for members, non-members and extra discount for winning mini game

Input age, gender, weight, height, activity level, daily goal, x, y, z

Output shop details, program features

*TSHouse*(tsHouse)

*Pogram*(pogram)

*Price1*(price1)

no

yes

Input update program

Output

- 3 guesses in mini game

- 5% discount for non-member

-8% discount if wins mini game

-6-month warranty for the program

Output

- 5 guesses in mini game

- 20% membership discount

-30% discount if wins mini game

-1-year warranty for the program

member

RETURN

Output :

-final price for member = (150 + 10) \* 0.8

-discount price for member = (150 + 10) \* 0.2

-final price for member winning mini game = (150 + 10) \* 0.7

-discount price for member winning mini game = (150 + 10) \* 0.3

-final price for non-member = 150\* 0.95

-discount price for non-member = 150\* 0.05

-final price for member winning mini game = 150 \* 0.92

-discount price for member winning mini game = 150 \* 0.08

-amount paid, cash = price

-change = cash – price

RETURN

Output RMR, TDEE, calories needed for losing, maintaining or gaining weight, recommended goal, maximum goal, speed, type of exercise, calories burned in one minute, calories yet to burn in order to achieve the daily goal, time needed to achieve goal

RETURN

Output voucher and receipt with shop name, address, careline, phone number, fax number, original price, discount price, discount rate, final price, the amount paid, the change and warranty code.

Payment succeeded

TRUE

FALSE

Amount paid >= price

TRUE

Transfer to cash payment

FALSE

for i > 3

Re-enter password

Payment succeeded

FALSE

TRUE

for 0 < i < 3,

Password 1 = password 2

Payment by card:

-Input numerical password by user

-Input numerical password again to confirm

Payment by cash:

-Input an amount to pay

Payment