Bukanga B

221005009

Input:

1. Age: The user inputs their age.

2. Heart Rate Values: The user provides 8 heart rate values.

Output:

1. Input Heart Rate Values: Displays the list of heart rate values entered by the user.

2. Heart Rate Percentages: Shows the calculated heart rate percentages based on the user's age.

3. Final Intensity: Indicates the workout intensity based on the average percentage of heart rate values.

Variables:

1. age (DWORD): Stores the user’s age.

2. inputArray (DWORD array, 8 elements): Stores the 8 heart rate values entered by the user.

3. outputArray (DWORD array, 8 elements): Stores the heart rate percentages calculated from the input values.

4. average (DWORD): Stores the average percentage of heart rate values.

5. maxHeartRate (DWORD): Stores the maximum heart rate calculated as 220 - age.

6. intensityMsg (BYTE array, 15 elements): Buffer to store the intensity message ("none", "light", "moderate", "vigorous").

7. Message Strings:

o msgPromptHR (BYTE): Prompt for heart rate value input.

o msgPromptAge (BYTE): Prompt for age input.

o msgColon (BYTE): Colon character for formatting.

o msgSpace (BYTE): Space character for formatting.

o msgNewLine (BYTE): Newline character for formatting.

o msgInputArray (BYTE): Label for displaying input heart rate values.

o msgOutputArray (BYTE): Label for displaying calculated heart rate percentages.

o msgFinalIntensity (BYTE): Label for displaying final intensity message.

o msgIntensityNone (BYTE): Intensity message for "none".

o msgIntensityLight (BYTE): Intensity message for "light".

o msgIntensityModerate (BYTE): Intensity message for "moderate".

o msgIntensityVigorous (BYTE): Intensity message for "vigorous".

o msgRightBracket (BYTE): Right bracket character for formatting.

o msgContinue (BYTE): Prompt asking if the user wants to continue.

Algorithm:

1. Initialize Program:

o Set pointers to inputArray and outputArray.

2. Prompt for Age:

o Display prompt for age.

o Read age input and store in age.

o Calculate maxHeartRate as 220 - age.

3. Input Heart Rate Values:

o Initialize a counter (ECX) to 0.

o Enter a loop to collect 8 heart rate values:

▪ Prompt user for heart rate value.

▪ Read heart rate input and store in inputArray.

▪ Calculate percentage as (heart_rate * 100) / maxHeartRate.

▪ Cap percentage at 100 if it exceeds 100.

▪ Store calculated percentage in outputArray.

4. Calculate Average Percentage:

o Initialize a counter (ECX) to 0 and accumulator (EAX) to 0.

o Enter a loop to sum values in outputArray.

o Calculate average percentage as the sum divided by 8.

5. Determine Intensity Level:

o Compare average percentage and classify intensity:

▪ If < 60%, set intensity to "none".

▪ If 60% <= average < 70%, set intensity to "light".

▪ If 70% <= average < 80%, set intensity to "moderate".

▪ If >= 80%, set intensity to "vigorous".

6. Display Results:

o Display input heart rate values.

o Display calculated heart rate percentages.

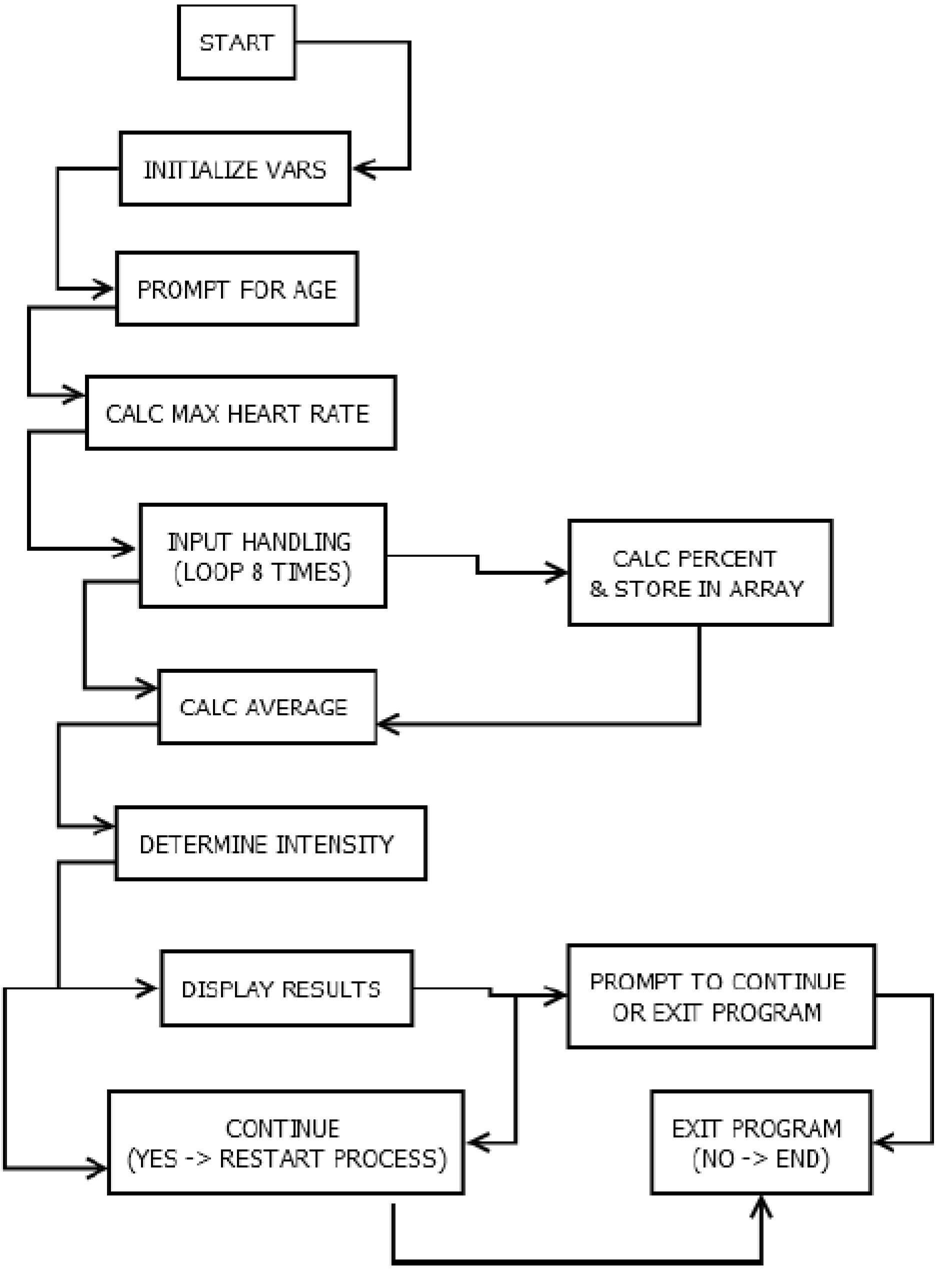
o Display final intensity message.

7. Continue or Exit:

o Prompt user to continue or exit.

o If user chooses to continue, restart from step 2.

o If user chooses to exit, terminate the program.


START
INITIALIZE VARS
PROMPT FOR AGE
CALC MAX HEART RATE
INPUT HANDLING (LOOP 8 TIMES)
CALC PERCENT & STORE IN ARRAY
CALC AVERAGE
DETERMINE INTENSITY
DISPLAY RESULTS
PROMPT TO CONTINUE OR EXIT PROGRAM
CONTINUE (YES -> RESTART PROCESS)
EXIT PROGRAM (NO -> END)