UNIVERSITI TUN HUSSEIN ONN MALAYSIA

SEMESTER 1 SESSION 2021/2022

GROUP PROJECT

BIC10204 : ALGORITHM & PROGRAMMING

ALPHA QUATTUOR (α-IV)

NAME		MATRIC NO
	MOHAMAD ARIF AZINUDDIN BIN ZAIDI	AI210125
	ABDUL ALIF BIN ABDUL GHAFOOR	AI210217

	**SITI AISHAH BINTI ABU BAKAR**	**AI210077**
	**AQMAL DANIAL BIN MUHAMMAD TAQIYUDIN**	**AI210181**

TABLE OF CONTENTS

1.0 PROJECT BACKGROUND

1.1) Title and theme

The project is titled ***Alpha Quattour (α-IV)*** because all member names in this group start with the letter *'A'*, thus, we decided to put the letter *'A'* for this project title but to make it unique, we use the *Greek* alphabet as we know that *alpha(α)* is the same as letter *'A'*. Next, the word is *'Quattuor'* and of that word is four, we use the *Latin* numeral because in this group, the members are only *4*.

1.2) Project objective

- To design a program that can calculate both the Basal Metabolic Rate (BMR) and Daily Calorie Requirement (DCR)
- To develop a program that can calculate and record both the Basal Metabolic Rate (BMR) and Daily Calorie Requirement (DCR) once a person key-in their gender, weight in kg, height in cm, and age in a year.
- To evaluate a program built so users can key in their information smoothly and get notified if they key in the wrong information.

2.0 TASK ASSIGNMENT

This section shows which part everyone is accountable for as everyone has a different field that they are good in. Throughout this project, we try to take advantage of each other's skills so the project objective can be achieved.

2.1) Group members

Key Members	**Person-in-charge**	**Contact information**
Programmer	Arif Azinuddin	011-1079 4886
Documentation 1	Siti Aishah	011-2402 3726
Documentation 2	Abdul Alif	014-234 1542
Presenter	Aqmal Danial	013-518 2395

2.2) Tasks

Programmer	● Brainstorm the solution to this question. ● Coding a C program based on the question. ● Mastermind of the project. ● Made a feedback that valid or invalid the input from the user. ● Invalid an alphabet character in int input section from the user. ● Ensure the output will be well arranged and systematically. ● Comment on a C program. ● Execute this project with high responsibility. ● Give a full commitment to this project and also be the leader. ● Track the group members' work progression. ● Compile the work from group members. ● Program code description. ● Output description. ● Prepare a slide.
Documentation 1	❖ Give ideas about the program. ❖ Brainstorm the objective of the project. ❖ Prepare a slide.

Documentation 2	★ Give comments and ideas about the program. ★ Write a pseudo-code for program development. ★ Prepare the slide for presentation. ★ Give full commitment to this project and presentation. ★ Assist group members. ★ Set-up time for group discussion . ★ Give an opinion about the C program.
Presenter	➔ Write a flowchart for program development. ➔ Present the project. ➔ Search reference.

3.0 PSEUDO-CODE AND FLOWCHART

Pseudocode is a simple code that resembles the program code, meanwhile a flowchart is an illustrated step for the programming process. In this section, we show both how the process of a program takes place and works from the first input until the end of the output.

3.1) PSEUDO-CODE

In this section, the statement is used as a guide for programmer in writing the code while keeping the writing process organized properly. This also helps in finding any mistake and ideas of improvement.

```
START
            display"/**************************************************
                                GROUP 2 PROJECT ALGORITHMS AND
                   PROGRAMMING(ARIF,ALIF,AQMAL,AISHAH)

                     ALPHA QUATTUOR (α-IV) coded & commented by Arif
    *************************************************************/"
            display" ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
            display"<                                                  >"
            display"< BMR & DCR CALCULATOR by ALPHA QUATTUOR  >"
            display"<                                                  >"
            display" ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"

            display"Enter Your Name (e.g : Aliff):"
            read name
            display"--Your Name is Valid--"

                set age = 1
                set ageRead = 0

              do
                display"Enter Your age (e.g : 20 years):"
                read age
                  While(ageRead != 1)
                      display"--That is not a number!! Please Try Again :) -- "
                      read %*[^\n]
                      display"Please insert a number for Age : "
                      read age
                  End while
                      if(age>0 && age<100)
                          display"-- Invalid Age !! Please Try Again :) --\n"
                      end if
             While(age<0 && age>100)
               display"-- Your Age is Valid :v --"

                      set weight = 1
                      set weightRead = 0
              do
                display"Enter Your Weight in kilogram (e.g : 70kg): "
                read weight
                  While(weightRead != 1)
```

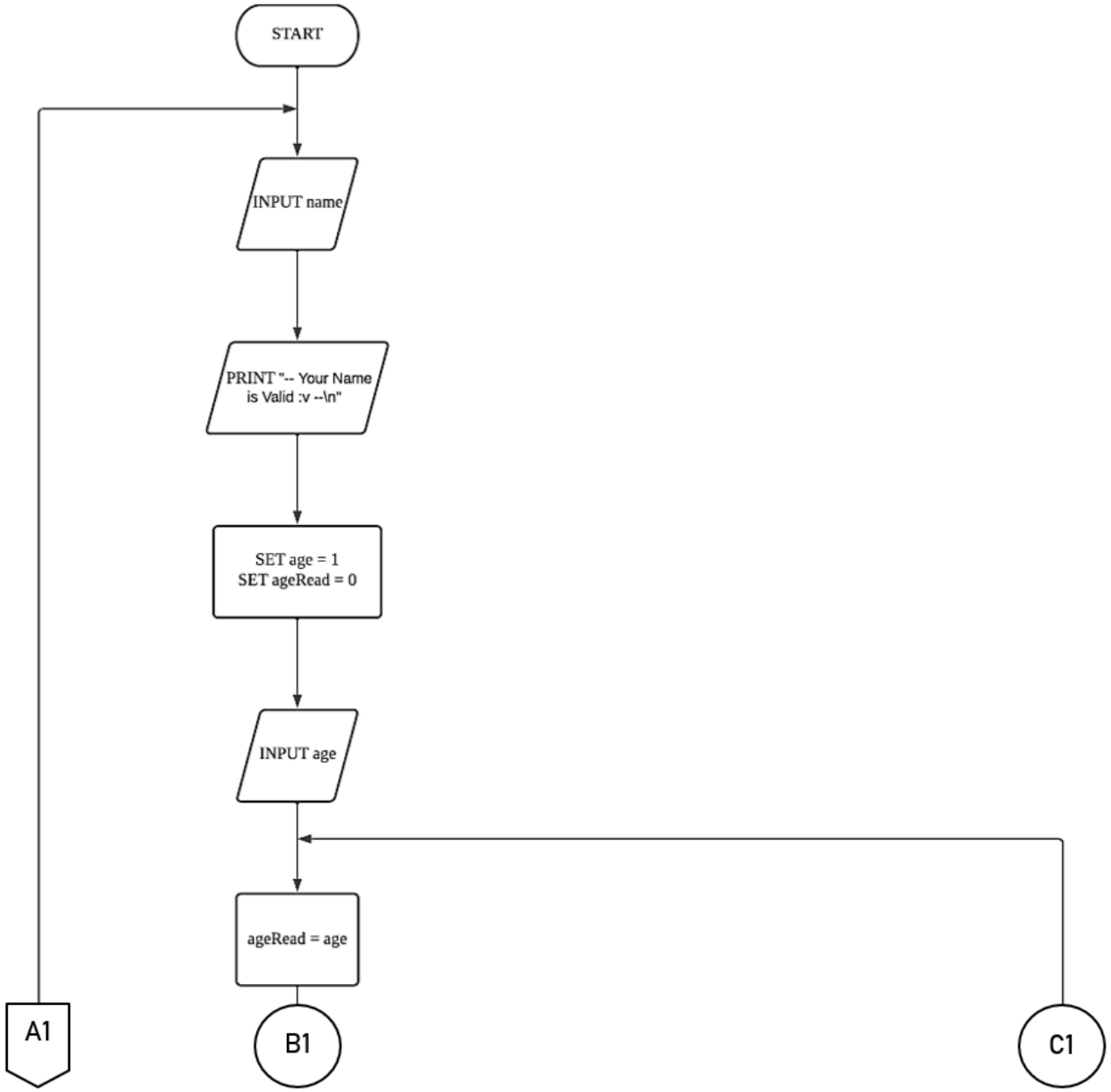
```
                display"--That is not a number!! Please Try Again :) -- "
                read %*[^\n]
                display"Please insert a number for Weight : "
                read weight
        End while
                if(weight<20 && weight>300)
                        display"-- Invalid Weight !! Please Try Again :) --"
                end if
While(weight<20 || weight>300)
                        display("-- Your Weight is Valid :v --\n");

                set height = 1
                set heightRead = 0
 do
      display"Enter Your Height in centimeter (e.g: 170cm): "
      read height
                While(heightRead != 1)
                 display"--That is not a number!! Please Try Again :) -- "
                 read %*[^\n]
                 display"Please insert a number for Height : "
                 read height

                End while
                if(height<90 && height>210)
                display"-- Invalid Height !! Please Try Again :) --"
                end if
     While (height<90 || height>210)
       display("-- Your Height is Valid :v --")

                set gender = 1
                set genderRead = 0
 do
      display"Enter Your Gender (e.g : 1 : Male or 2 : Female): "
      read gender
        While(genderRead != 1)
                display"--That is not a number!! Please Try Again :) -- "
                read %*[^\n]
                display "Please insert a number for Gender : "
                read gender
 End while
                if (gender=1)
                BMR = 66+(13.7*weight)+(5*height)-(6.8*age
                display" -- Your Gender is Valid :v --"
                display "Male"
                else if(gender=2)
                BMR = 655+(9.6*weight)+(1.8*height)-(4.7*age)
                printf(" -- Your Gender is Valid :v --"
                display "Female"
                else
                display"-- Invalid Gender !! Please Try Again :) --"
                end if
       while(gender<1 || gender>2)
```

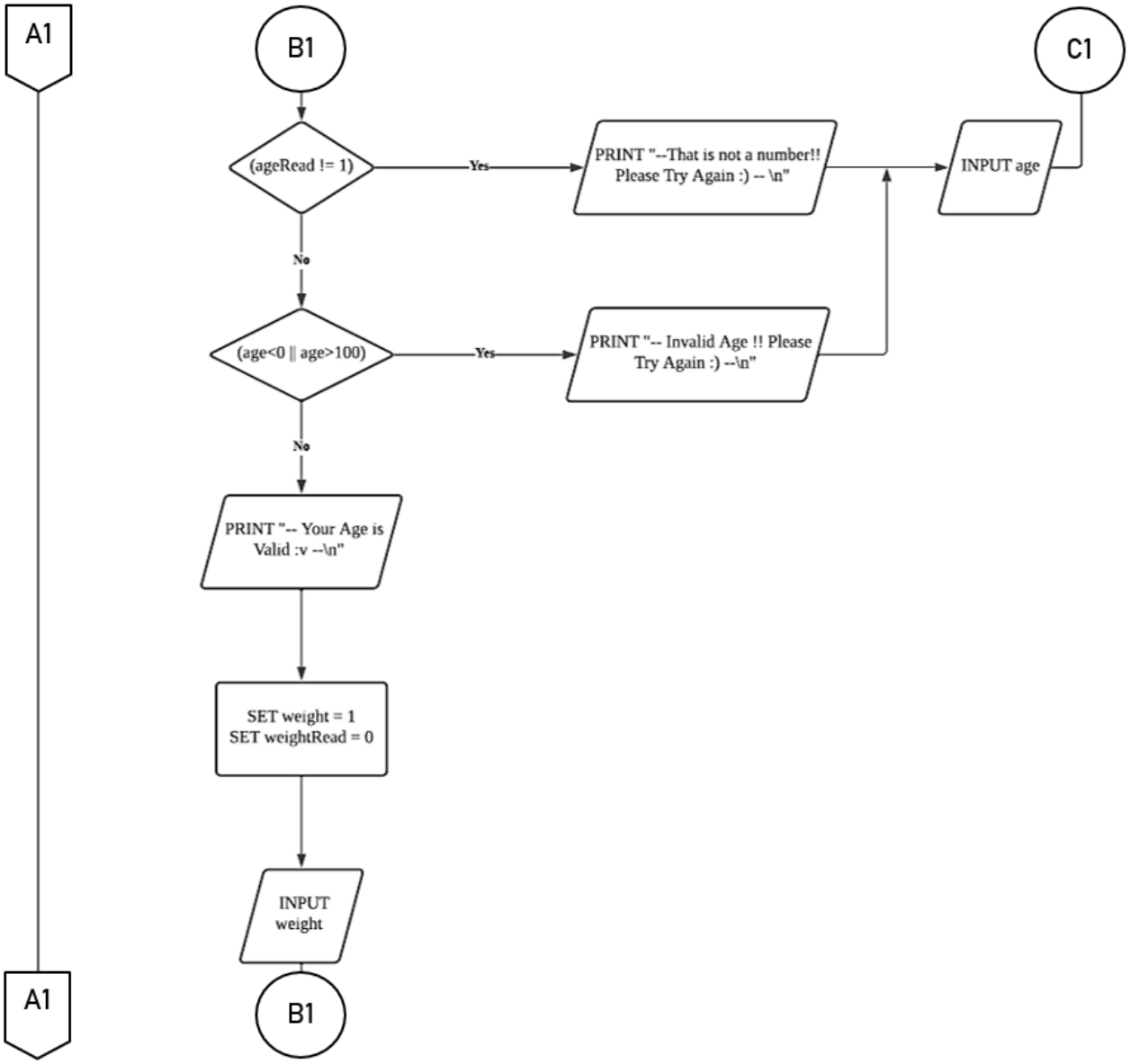
```
                set level = 1
                set readlevel = 0
    do
        display "Enter Your Level of Activity: "
        display "1. Sedentary : little or no exercise, desk job
        2.Light Activity : light exercise or sports 1-3 days/week 3.
        Moderate Activity : moderate exercise or sports 3-5 days/week4.
        Very Active : hard exercise or sports 6-7 days/week5.
        Extra Active : hard daily exercise or sports & physical job
        or 2x day training, i.e. marathon, contest. Enter your choice: "

    while (levelRead != 1)
            display"--That is not a number!! Please Try Again :) -- "
            read %*[^\n]
            display "Please insert a number for Level of Activity : "
            read level
     End while
            if(level>=1 && level<=5)
                display" -- Your Level of Activity is Valid :v --"
            else
                display"-- Invalid Level of Activity!! Please Try Again
                    :) --"
            end if

          case based on number
            case 1:
              DCR = BMR*1.2;
              strcpy"Sedentary"

            case 2:
              DCR = BMR*1.375;
              display"Light Activity"

            case 3:
              DCR = BMR*1.55;
              display"Moderate Activity"

            case 4:
              DCR = BMR*1.725;
              display"Very Active"

            case 5:
              DCR = BMR*1.9;
              display"Extra Active"

            default:
              DCR = 0;
              display"Invalid Activity"
                End case
display"----------------------------Summary-----------------------------"
display name
display age
display weight
```

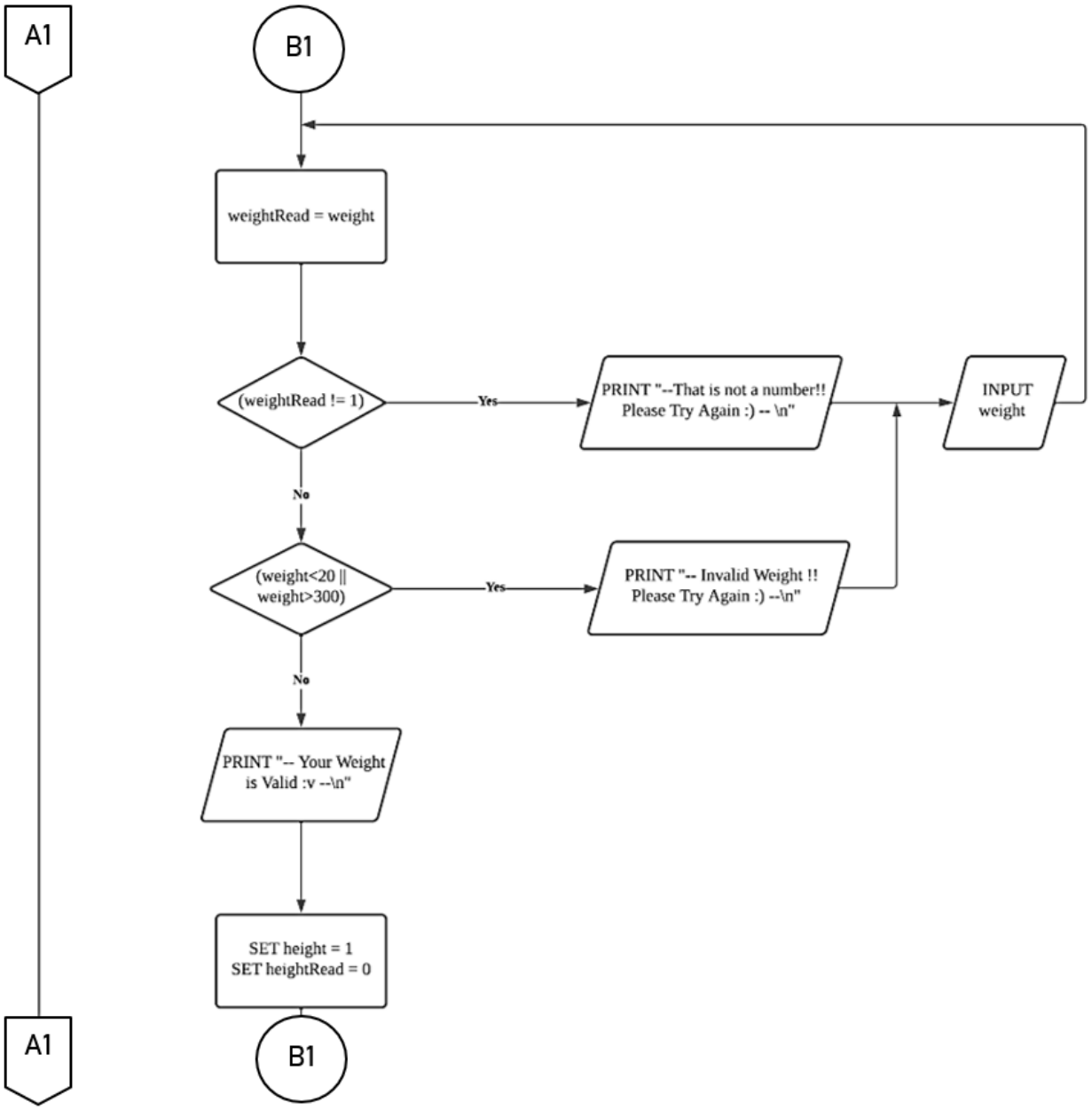
```
display height
display sex
display activity
display calculation for BMR
display BMR
display calculation for DCR
display DCR
display "Note Based on this calculation, you would need DCR
        calories food to sustain your daily activities."
display"==============THANK YOU================"
        set end_task = 1
        set end_taskRead = 0
    do
        display"Press '1' if you want to re-enter the input and
                Press '2' to end this program :) : "
        read end_task
    while (end_taskRead != 1)
        display"--That is not a number!! Please Try Again :) -- "
        read %*[^\n]
        display "Please insert a number for this question: "
        read end_task
End while

      if(end_task==1) goto loop

      else if(end_task==2)
              display"========END OF PROGRAM=========="\
          break
      else
              display"-- Invalid Number !! Please Try Again :) --"

While(end_task<1 || end_task>2)
STOP
```

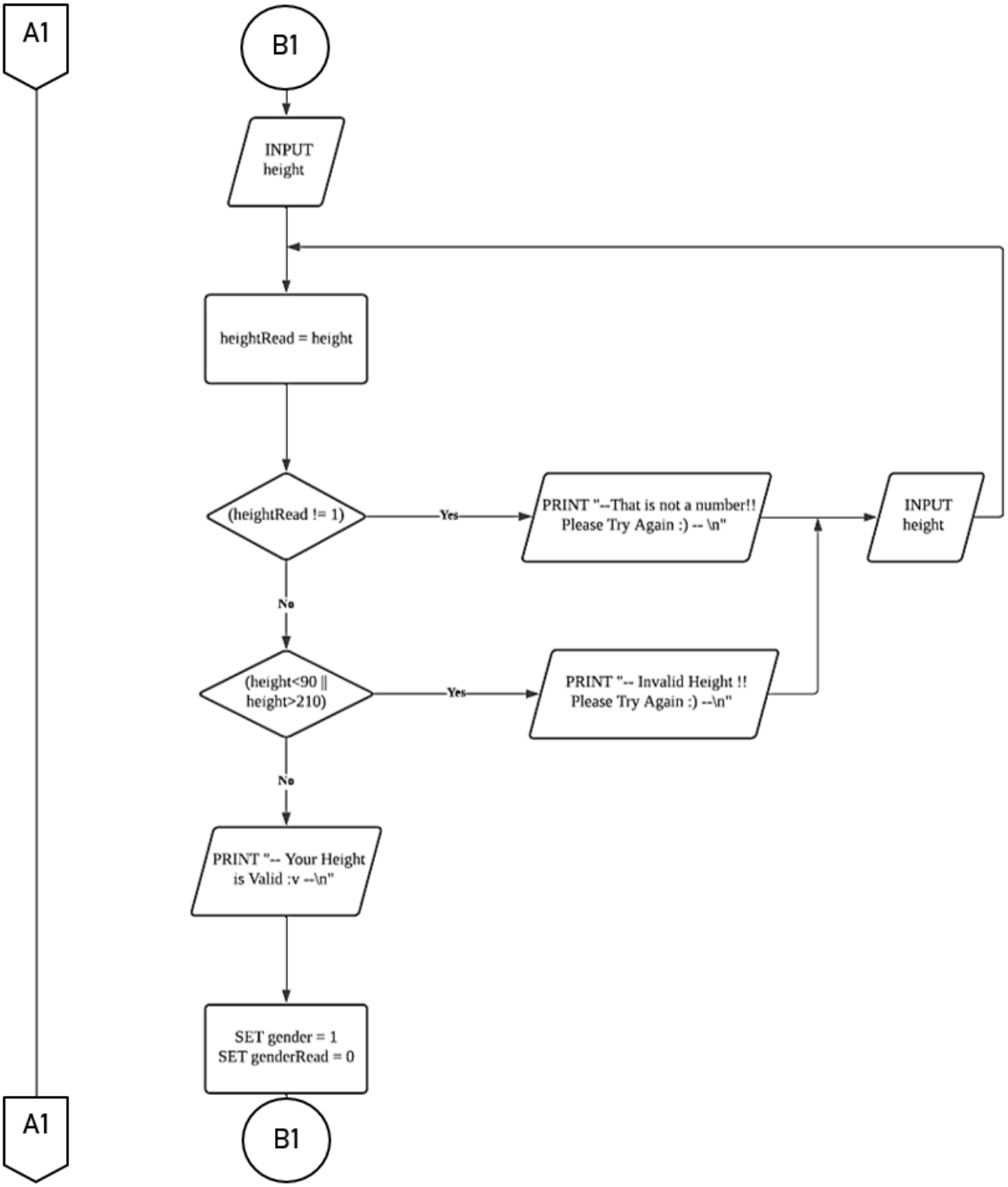
3.2)FLOWCHART

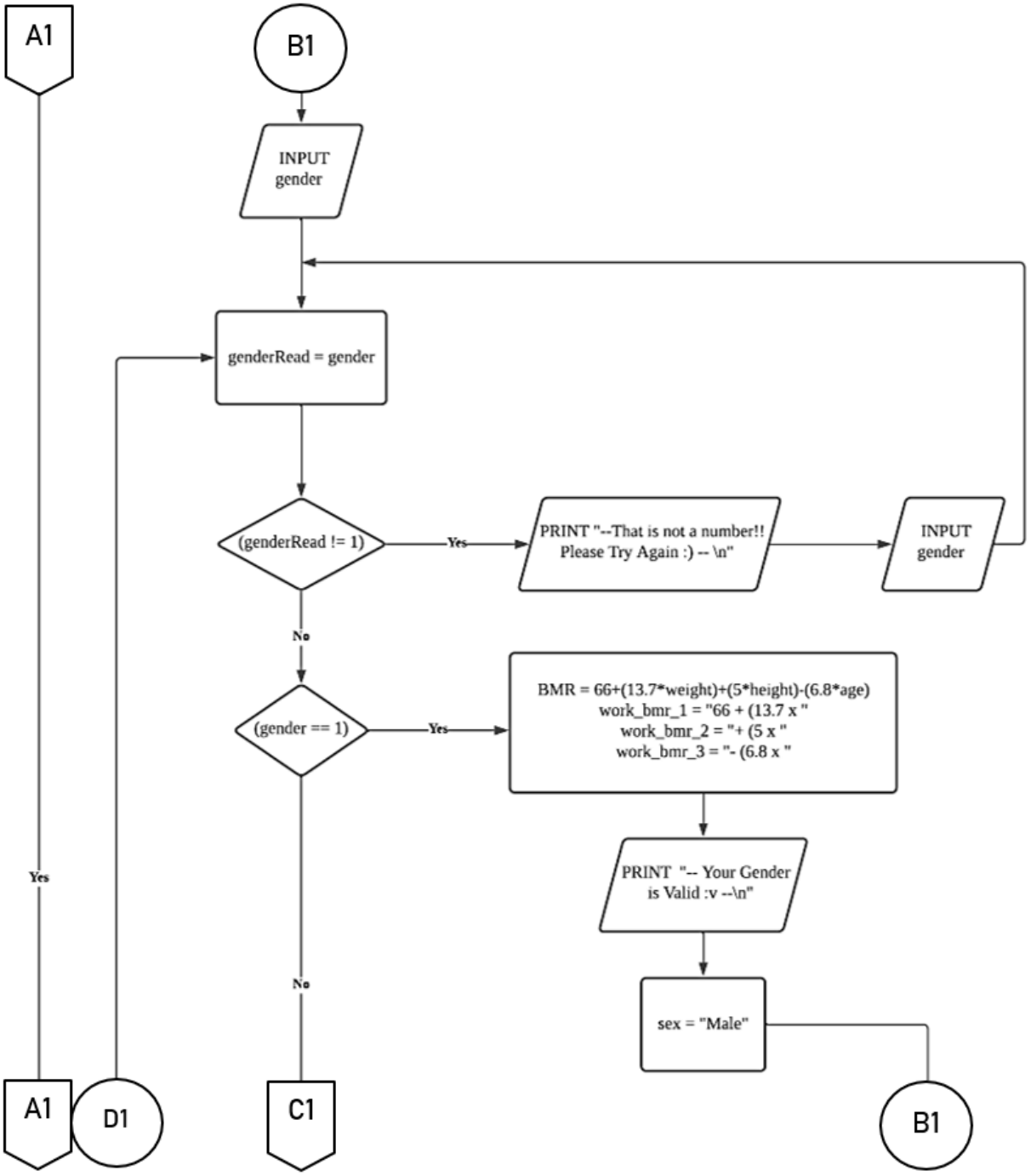
In this section we can see the flow of work the program will run through and the flows it takes to choose the right output based on the data input.

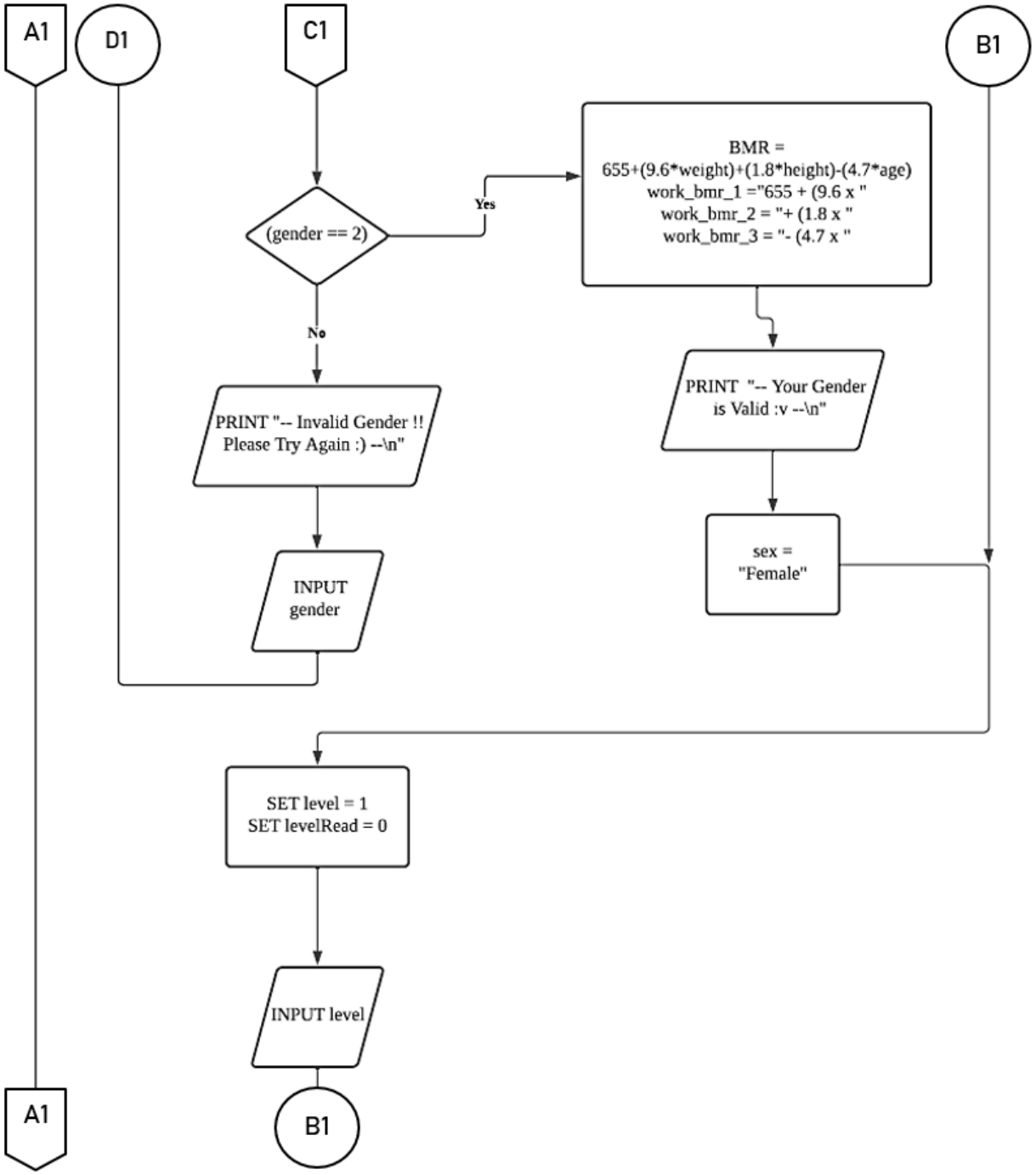


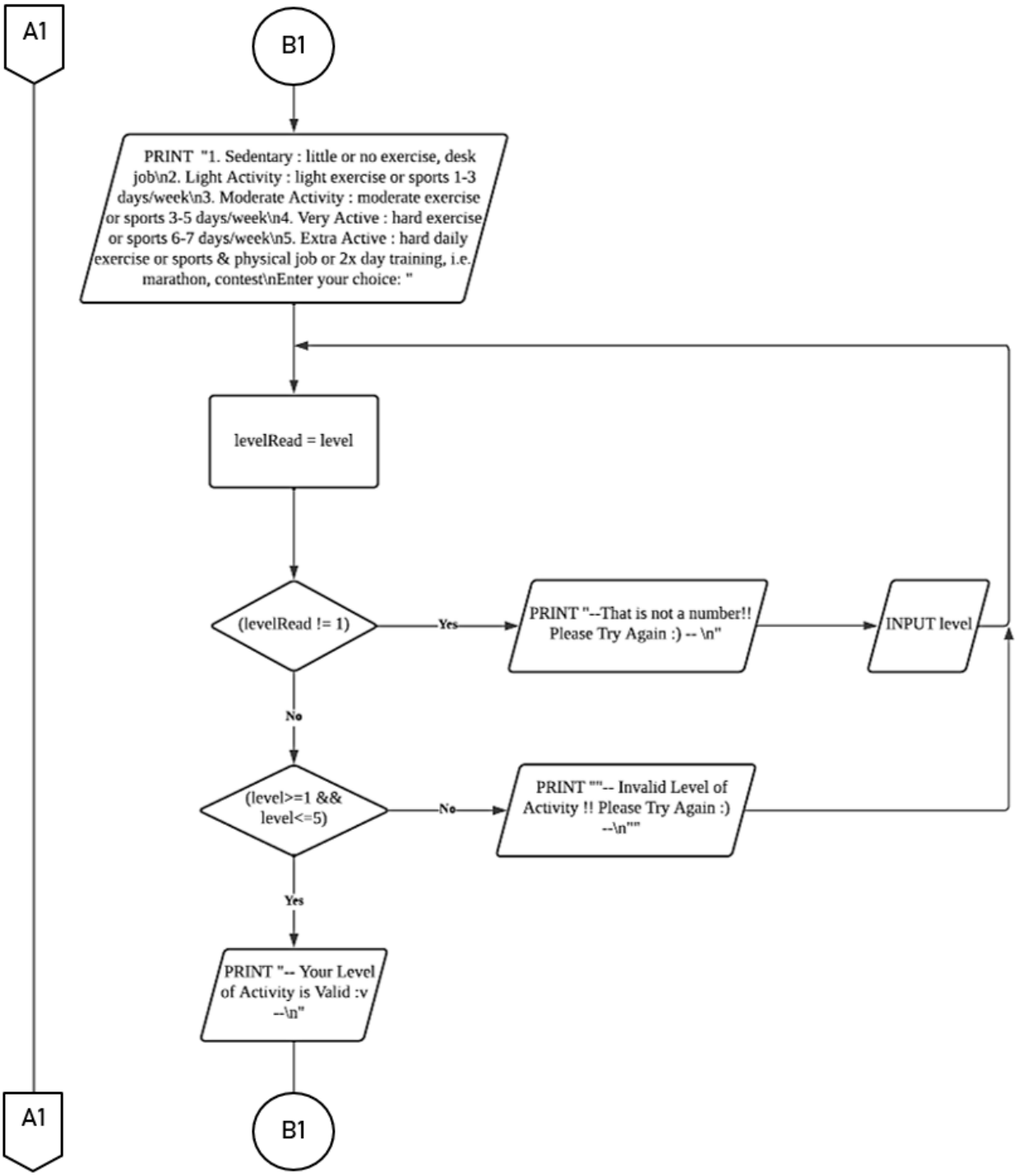

A1
B1
C1
(ageRead != 1)
Yes
PRINT "--That is not a number!! Please Try Again :) -- \n"
INPUT age
No
(age<0 || age>100)
Yes
PRINT "-- Invalid Age !! Please Try Again :) --\n"
No
PRINT "-- Your Age is Valid :v --\n"
SET weight = 1
SET weightRead = 0
INPUT weight
A1
B1

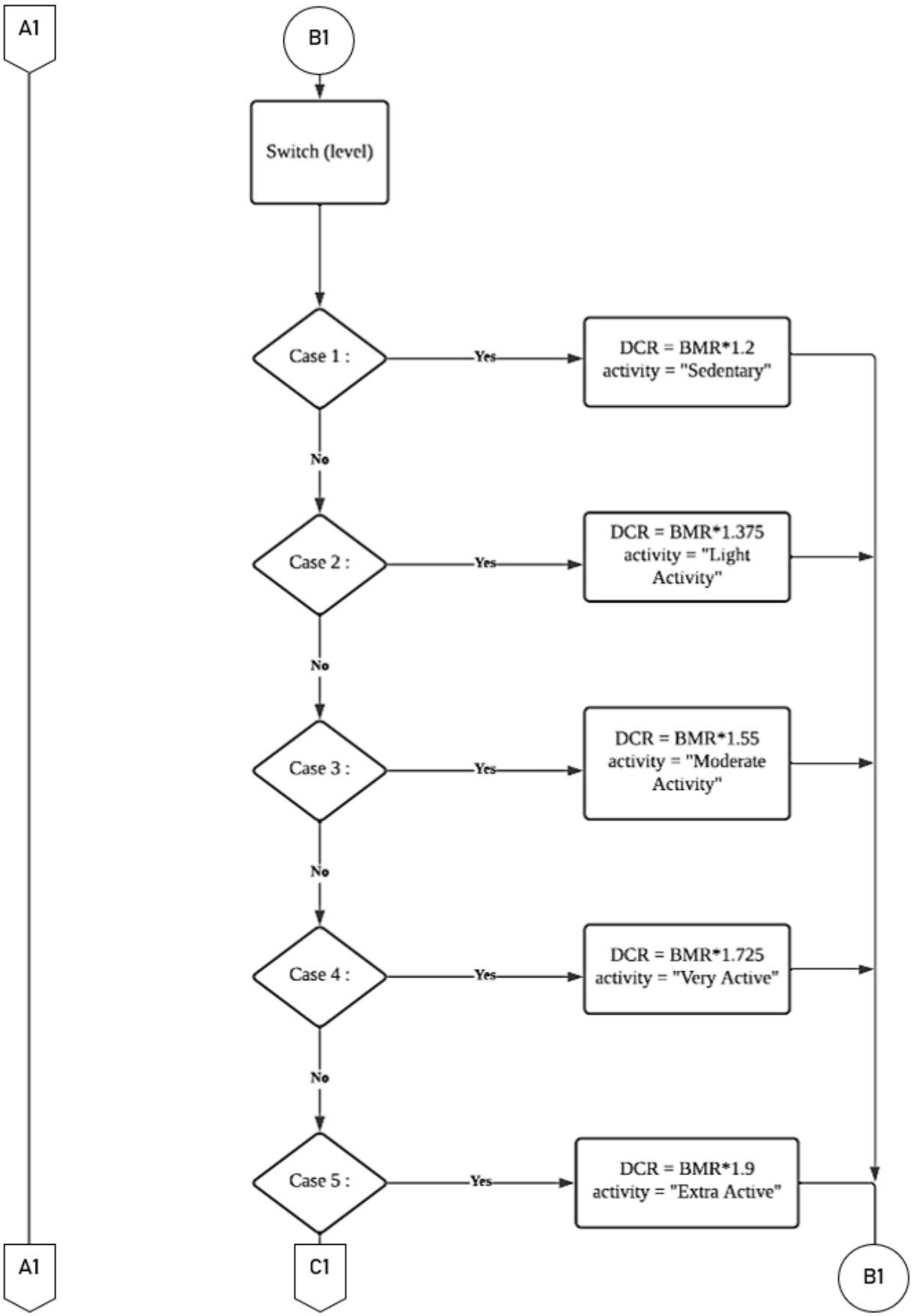

A1
B1
weightRead = weight
(weightRead != 1)
Yes
PRINT "--That is not a number!!
Please Try Again :) -- \n"
INPUT
weight
No
(weight<20 ||
weight>300)
Yes
PRINT "-- Invalid Weight !!
Please Try Again :) --\n"
No
PRINT "-- Your Weight
is Valid :v --\n"
SET height = 1
SET heightRead = 0
B1
A1

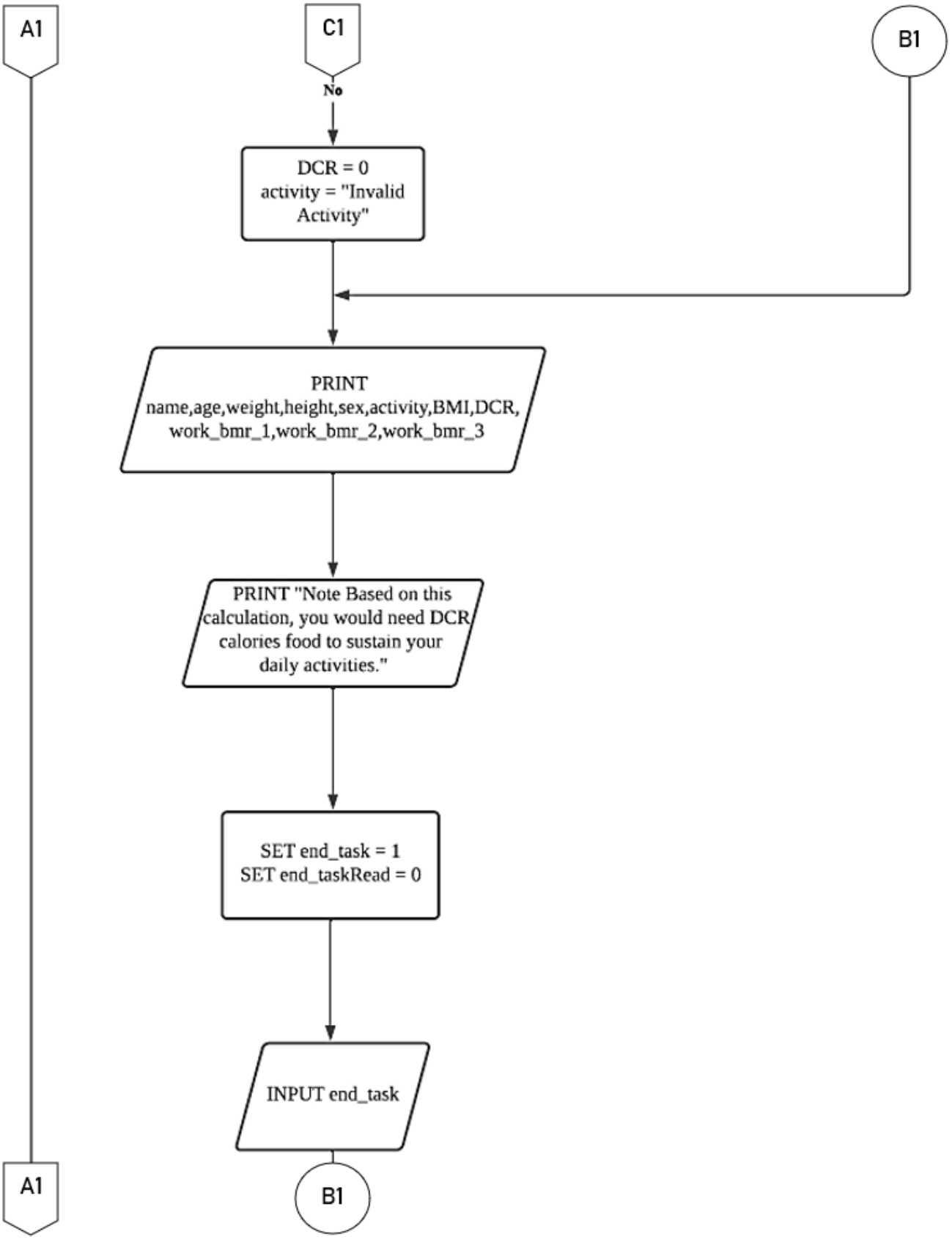

A1
B1
INPUT
height
heightRead = height
(heightRead != 1)
Yes
PRINT "--That is not a number!!
Please Try Again :) -- \n"
INPUT
height
No
(height<90 ||
height>210)
Yes
PRINT "-- Invalid Height !!
Please Try Again :) --\n"
No
PRINT "-- Your Height
is Valid :v --\n"
SET gender = 1
SET genderRead = 0
A1
B1

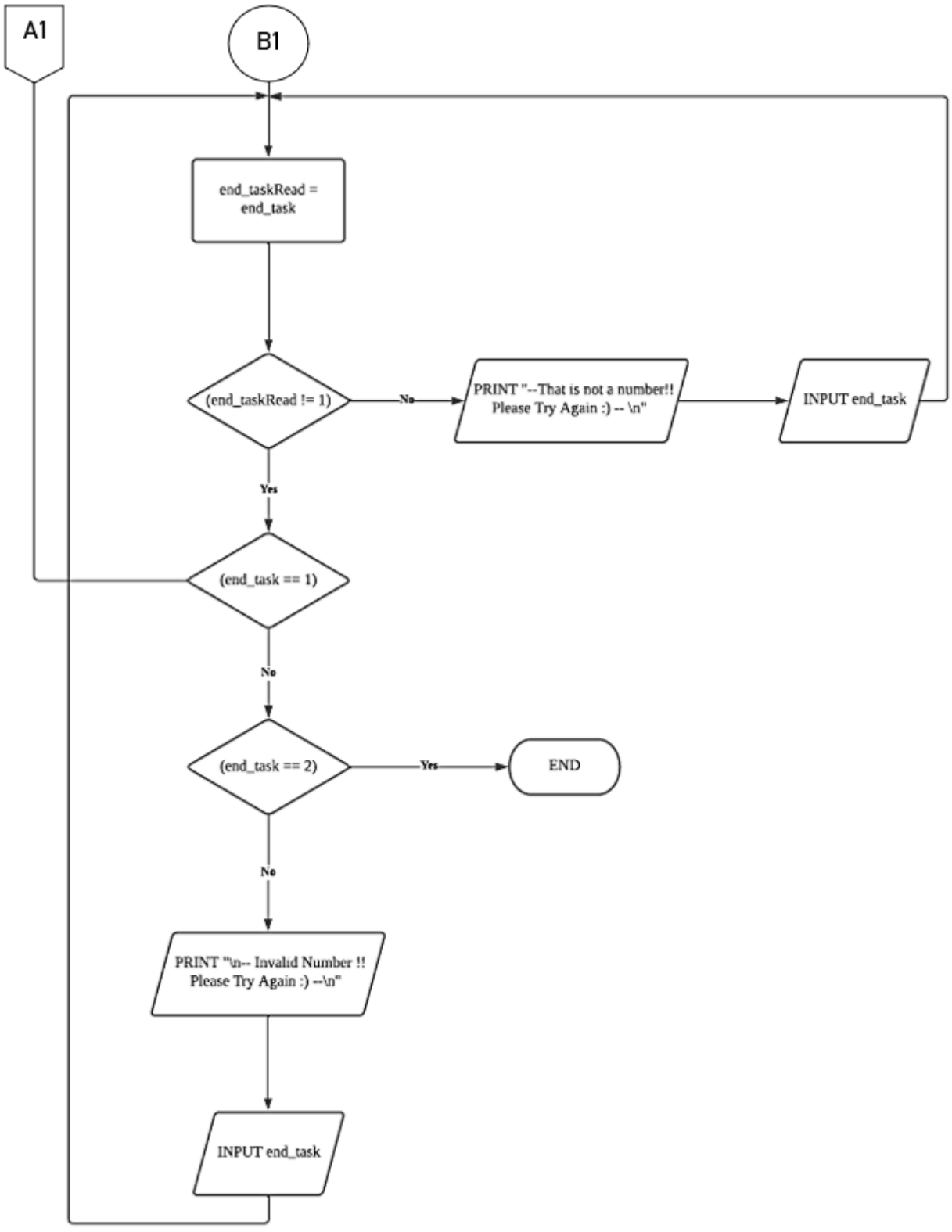

A1
B1
INPUT
gender
genderRead = gender
(genderRead != 1)
Yes
PRINT "--That is not a number!!
Please Try Again :) -- \n"
INPUT
gender
No
(gender == 1)
Yes
BMR = 66+(13.7*weight)+(5*height)-(6.8*age)
work_bmr_1 = "66 + (13.7 x "
work_bmr_2 = "+ (5 x "
work_bmr_3 = "- (6.8 x "
PRINT "-- Your Gender
is Valid :v --\n"
sex = "Male"
Yes
No
A1
D1
C1
B1


A1
D1
C1
B1
(gender == 2)
Yes
No
BMR =
655+(9.6*weight)+(1.8*height)-(4.7*age)
work_bmr_1 ="655 + (9.6 x "
work_bmr_2 = "+ (1.8 x "
work_bmr_3 = "- (4.7 x "
PRINT "-- Invalid Gender !!
Please Try Again :) --\n"
PRINT "-- Your Gender
is Valid :v --\n"
INPUT
gender
sex =
"Female"
SET level = 1
SET levelRead = 0
INPUT level
A1
B1


A1
B1
PRINT "1. Sedentary : little or no exercise, desk job\n2. Light Activity : light exercise or sports 1-3 days/week\n3. Moderate Activity : moderate exercise or sports 3-5 days/week\n4. Very Active : hard exercise or sports 6-7 days/week\n5. Extra Active : hard daily exercise or sports & physical job or 2x day training, i.e. marathon, contest\nEnter your choice: "
levelRead = level
(levelRead != 1)
Yes
PRINT "--That is not a number!! Please Try Again :) -- \n"
INPUT level
No
(level>=1 && level<=5)
No
PRINT ""-- Invalid Level of Activity !! Please Try Again :) --\n""
Yes
PRINT "-- Your Level of Activity is Valid :v --\n"
A1
B1


A1
B1
Switch (level)
Case 1 :
Yes
DCR = BMR*1.2
activity = "Sedentary"
No
Case 2 :
Yes
DCR = BMR*1.375
activity = "Light
Activity"
No
Case 3 :
Yes
DCR = BMR*1.55
activity = "Moderate
Activity"
No
Case 4 :
Yes
DCR = BMR*1.725
activity = "Very Active"
No
Case 5 :
Yes
DCR = BMR*1.9
activity = "Extra Active"
A1
C1
B1


A1
C1
No
B1
DCR = 0
activity = "Invalid Activity"
PRINT
name,age,weight,height,sex,activity,BMI,DCR,
work_bmr_1,work_bmr_2,work_bmr_3
PRINT "Note Based on this calculation, you would need DCR calories food to sustain your daily activities."
SET end_task = 1
SET end_taskRead = 0
INPUT end_task
B1
A1


A1
B1
end_taskRead = end_task
(end_taskRead != 1)
No
PRINT "--That is not a number!! Please Try Again :) -- \n"
INPUT end_task
Yes
(end_task == 1)
No
(end_task == 2)
Yes
END
No
PRINT "\n-- Invalid Number !! Please Try Again :) --\n"
INPUT end_task

4.0 PROGRAM CODE DESCRIPTION

```c
/*****************************************************************************
      GROUP 2 PROJECT ALGORITHMS AND PROGRAMMING(ARIF,ALIFF,AQMAL,AISHAH)

                 ALPHA QUATTUOR (α-IV) coded & commented by Arif
*****************************************************************************/
//declare header
#include <stdio.h>
#include <string.h>
#include <math.h>
```

Figure 1: *Headers used in the program.*

Arif declared header files due to the fact that it is possible to have multiple header files that perform various tasks. This header file allows you to do console input and output actions.

```c
//declare function
int main()
{
```

Figure 2: *Main function used in the program.*

Arif declared the function of main() to specify the starting place of your programme, where the C source code you write will be located in the program. This is what a function declaration informs the compiler. It specifies what the function name is and how to execute it.

```c
        //declare identifier data types
        char name[50];
        char activity[20];
        char sex[10];
        int BMR, DCR;
```

Figure 3: *Declaration made in the program.*

A unique name is given to an entity so that it could be identified during execution of the programme. Theoretically, identifiers are made to do this. Thus, Arif declared the data types of identifiers in this programming. For example, char for name, activity, and sex; it receives alphabet from user. Next, int BMR and DCR only receive an input number from the user.

```c
        //label statement for goto function
        again:
```

Figure 4: *Label statement.*

Arif declared a label again: just for the user to re-enter the input without ending this programming. Thus, this is looping programming.

```c
        //display header for user
        printf("\n ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~");
        printf("\n<                                                              >");
        printf("\n<\tBMR & DCR CALCULATOR by ALPHA QUATTUOR :v\t  >");
        printf("\n<                                                              >");
        printf("\n ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~\n");
```

Figure 5: *Display header for user.*

This is just a header of this programming that the user can see when running the programming.

```
30          //input from user
31          //user enter their name
32          printf("\nEnter Your Name (e.g : Aliff): ");
33          scanf("%s",name);
34
35          //feedback to user
36          printf("-- Your Name is Valid :v --\n");
```

Figure 6: *Input name from users.*

This is the starting section of input from the user. Printf function is used to display a string on the screen and the user can understand what they need to do after reading the question. Scanf function is used to read the input data from the user and the program can print the output same as the data that the user key-in.

```
38          //declare age for receive number input and ageRead for invalid alphabet
39          int age = 1;
40          int ageRead = 0;
41
42          //(loop do-while-while)if-else statement for age
43          do
44          {
45          printf("Enter Your Age (e.g : 20 years): ");
46          ageRead = scanf("%d", &age);
47
48          while (ageRead != 1)
49          {
50              printf("--That is not a number!! Please Try Again :) -- \n");
51              scanf("%*[^\n]");
52              printf("Please insert a number for Age : ");
53              ageRead = scanf("%d", &age);
54          }
55
56              //if-else statement for age
57              if(age<0 || age>100)
58                  {
59                      printf("-- Invalid Age !! Please Try Again :) --\n");
60                  }
61          }
62          while(age<0 || age>100);
63
64          //feedback to user
65          printf("-- Your Age is Valid :v --\n");
```

Figure 7: *Age input from user.*

Arif declared int age = 1 to receive input data from the user and int ageRead = 0 to invalid an alphabet input from the user. If the condition ageRead !=1 is obeyed, it will loop until the user inserts the input of a number. The %*[^\n] function is to scan everything until a newline \n (does not scan \n) and discards it, that is does not store the scanned data anywhere. If-else statement is used to determine the validation of an input from the user. Thus, if the user inserts -1 for age, it will loop until the user has the valid input. The valid range for age is from 0 to 100. After the user's input is valid, the program will give feedback to the user and the program will proceed to the weight section.

```
71          //(loop do-while-while)if-else statement for weight
72          do
73          {
74              printf("Enter Your Weight in kilogram (e.g : 70kg): ");
75              weightRead = scanf("%d",&weight);
76
77          while (weightRead != 1)
78          {
79              printf("--That is not a number!! Please Try Again :) -- \n");
80              scanf("%*[^\n]");
81              printf("Please insert a number for Weight : ");
82              weightRead = scanf("%d", &weight);
83          }
84
85          //if-else statement for weight
86          if(weight<20 || weight>300)
87              {
88                  printf("-- Invalid Weight !! Please Try Again :) --\n");
89              }
90          }
91          while(weight<20 || weight>300);
92
93          //feedback to user
94          printf("-- Your Weight is Valid :v --\n");
```

Figure 8: *Weight input from user.*

Next, Arif declared int weight = 1 to receive input data from the user and int weightRead = 0 to invalid an alphabet input from the user. If the condition is weightRead !=1 is obeyed, it will loop until the user inserts the input of a number. The %*[^\n] function is to scan everything until a newline \n (does not scan \n) and discards it, that is does not store the scanned data anywhere. If-else statement is used to determine the validation of an input from the user. Thus, if the user inserts 19 for weight, it will loop until the user has the valid input. The valid range for weight is from 20 to 300. After the user's input is valid, the program will give feedback to the user and the program will proceed to the height section.

```
96          //declare height for receive number input and heightRead for invalid alphabet
97          int height = 1;
98          int heightRead = 0;
99
100         //(loop do-while-while)if-else statement for height
101         do
102         {
103         printf("Enter Your Height in centimeter (e.g: 170cm): ");
104         heightRead = scanf("%d",&height);
105
106         while (heightRead != 1)
107         {
108             printf("--That is not a number!! Please Try Again :) -- \n");
109             scanf("%*[^\n]");
110             printf("Please insert a number for Height : ");
111             heightRead = scanf("%d", &height);
112         }
113
114         //if-else statement for height
115         if(height<90 || height>210)
116             {
117                 printf("-- Invalid Height !! Please Try Again :) --\n");
118             }
119         }
120         while(height<90 || height>210);
121
122         //feedback to user
123         printf("-- Your Height is Valid :v --\n");
```

Figure 9: *Height input from user.*

Arif declared int height = 1 to receive input data from the user and int heightRead = 0 to invalid

an alphabet input from the user. If the condition is heightRead !=1 is obeyed, it will loop until the user inserts the input of a number. The %*[^\n] function is to scan everything until a newline \n (does not scan \n) and discards it, that is does not store the scanned data anywhere. If-else statement is used to determine the validation of an input from the user. Thus, if the user inserts 211 for height, it will loop until the user has the valid input. The valid range for age is from 90 to 210. After the user's input is valid, the program will give feedback to the user and the program will proceed to the gender section.

```
125         //declare gender for receive number input and genderRead for invalid alphabet
126         int gender = 1;
127         int genderRead = 0;
128         //for bmr working step output
129         char work_bmr_1[50];
130         char work_bmr_2[50];
131         char work_bmr_3[50];
132
133         //(loop do-while-while)if-else statement for gender
134         do
135         {
136         printf("Enter Your Gender (e.g : 1 : Male or 2 : Female): ");
137         genderRead = scanf("%d", &gender);
138
139         while (genderRead != 1)
140         {
141             printf("--That is not a number!! Please Try Again :) -- \n");
142             scanf("%*[^\n]");
143             printf("Please insert a number for Gender : ");
144             genderRead = scanf("%d", &gender);
145         }
146
147         //if-else statement for gender
148         if(gender==1)
149                 {
150                     BMR = 66+(13.7*weight)+(5*height)-(6.8*age);
151                     strcpy(work_bmr_1,"66 + (13.7 x ");
152                     strcpy(work_bmr_2,"+ (5 x ");
153                     strcpy(work_bmr_3,"- (6.8 x ");
154                     printf("-- Your Gender is Valid :v --\n");
155                     strcpy(sex,"Male");
156                 }
157         else if(gender==2)
158                 {
159                     BMR = 655+(9.6*weight)+(1.8*height)-(4.7*age);
160                     strcpy(work_bmr_1,"655 + (9.6 x ");
161                     strcpy(work_bmr_2,"+ (1.8 x ");
162                     strcpy(work_bmr_3,"- (4.7 x ");
163                     printf("-- Your Gender is Valid :v --\n");
164                     strcpy(sex,"Female");
165                 }
166         else
167                 {
168                     printf("-- Invalid Gender !! Please Try Again :) --\n");
169                 }
170         }
171         while(gender<1 || gender>2);
```

Figure 10: *Gender input from user.*

Arif declared int gender = 1 to receive input data from the user and int genderRead = 0 to invalid an alphabet input from the user. Arif also declared work_bmr_1, work_bmr_2 and work_bmr_3 for output purposes. If the condition is genderRead !=1 is obeyed, it will loop until the user inserts the input of a number. The %*[^\n] function is to scan everything until a newline \n (does not scan \n) and discards it, that is does not store the scanned data anywhere. If-else statement is used to determine the validation of an input from the user. If the user input number 1, the program will calculate the BMR of male using the formula, Arif used strcpy function for output purpose and the program gives the feedback to the user. After that, if the user input number 2, the program will calculate the BMR of female using the formula, Arif also used strcpy function for output purpose and the program gives the feedback to the user. Thus, if the user inserts 0 for gender, it will loop until the user has the valid input. The valid range for age is from 1 to 2. After the user's input is valid, the program will proceed to the level of activity section.

```
173         //declare level for receive number input and levelRead for invalid alphabet
174         int level = 1;
175         int levelRead = 0;
176 
177         //(loop do-while-while)if-else statement for level
178         do
179         {
180             printf("Enter Your Level of Activity: \n");
181             printf("1. Sedentary : little or no exercise, desk job\n2. Light Activity : 1
182             levelRead = scanf("%d", &level);
183 
184         while (levelRead != 1)
185         {
186             printf("--That is not a number!! Please Try Again :) -- \n");
187             scanf("%*[^\n]");
188             printf("Please insert a number for Level of Activity : ");
189             levelRead = scanf("%d", &level);
190         }
191 
192             //if-else statement for level of activity
193             if(level>=1 && level<=5)
194                     {
195                         printf("-- Your Level of Activity is Valid :v --\n");
196                     }
197             else
198                     {
199                     printf("-- Invalid Level of Activity !! Please Try Again :) --\n");
200                     }
201 
202         }
203         while(level<1 || level>5);
204 
205         char work_dcr[10];
```

Figure 11: *Level of activity input from user.*

Arif declared int level = 1 to receive input data from the user and int levelRead = 0 to invalid an alphabet input from the user. If the condition is levelRead !=1 is obeyed, it will loop until the user inserts the input of a number. The %*[^\n] function is to scan everything until a newline \n (does not scan \n) and discards it, that is does not store the scanned data anywhere. If-else statement is used to determine the validation for level of activity which was input from the user. If the user input number 1 until 5, the program will give the feedback to the user. Thus, if the user inserts 0 for level of activity, it will loop until the user has the valid input. The valid range for age is from 1 to 5. After the user's input is valid, the program will proceed to the switchcase for the DCR calculation section.

```
205         //for dcr working step output
206         char work_dcr[10];
207
208         //switchcase for DCR calculation
209         switch(level)
210         {
211                 case 1:
212                         DCR = BMR*1.2;
213                         strcpy(work_dcr,"1.2");
214                         strcpy(activity,"Sedentary");
215                         break;
216                 case 2:
217                         DCR = BMR*1.375;
218                         strcpy(work_dcr,"1.375");
219                         strcpy(activity,"Light Activity");
220                         break;
221                 case 3:
222                         DCR = BMR*1.55;
223                         strcpy(work_dcr,"1.55");
224                         strcpy(activity,"Moderate Activity");
225                         break;
226                 case 4:
227                         DCR = BMR*1.725;
228                         strcpy(work_dcr,"1.725");
229                         strcpy(activity,"Very Active");
230                         break;
231                 case 5:
232                         DCR = BMR*1.9;
233                         strcpy(work_dcr,"1.9");
234                         strcpy(activity,"Extra Active");
235                         break;
236                 default:
237                         DCR = 0;
238                         strcpy(activity,"Invalid Activity");
239                         break;
240         }
```

Figure 12: *Switchcase for DCR calculation.*

Arif declared char work_dcr for output purpose. Next, this switchcase is the consequence from gender and level of activity section. BMR value received from the gender section and every level of activity have their own coefficient to be multiplied with BMR value. Arif used strcpy function (activity) for output purpose. This is the last execution section in this program.

```
242         //output for user
243         printf("\n\n----------------------------Summary-----------------------------");
244         printf("\nName\t: %s",name);
245         printf("\nAge\t: %d years",age);
246         printf("\nWeight\t: %d kg",weight);
247         printf("\nHeight\t: %d cm",height);
248         printf("\nGender\t: %s",sex);
249         printf("\nLevel of Activity : %s",activity);
250         printf("\nBMR\t: Calculation = %s %d kg) %s %d cm) %s %d years)", work_bmr_1, weight, work_bmr_2, height, work_bmr_3, age);
251         printf("\nBMR\t: %d calories",BMR);
252         printf("\nDCR\t: Calculation = %d calories x %s", BMR, work_dcr);
253         printf("\nDCR\t: %d calories",DCR);
254         printf("\nNote\t: Based on this calculation, you would need %d calories \n \t  food to sustain your daily activities.",DCR);
255
256         printf("\n\n=========================THANK YOU============================\n\n");
```

Figure 13: *Output for user.*

Output section is for printing all of the input from the user. Name, age, weight, height, gender, level of activity, working step calculation for BMR, BMR, working step calculation for DCR DCR and the note will be displayed in the output section. The strcpy for gender will print “Male” and “Female” instead of the number 1 and 2. For level of activity also strcpy will print “Sedentary”, “Light Activity”, “Moderate Activity”, “Very Active”, and “Extra Active” instead of the number 1, 2, 3, 4, and 5.

```
        //declare end_task for receive number input and end_taskRead for invalid alphabet
        int end_task = 1;
        int end_taskRead = 0;

        //(loop do-while-while)if-else statement for end_task
        do
        {
        printf("\nPress '1' if you want to re-enter the input and \nPress '2' to end this program :) : ");
        end_taskRead = scanf("%d", &end_task);

        while (end_taskRead != 1)
            {
                printf("--That is not a number!! Please Try Again :) -- \n");
                scanf("%*[^\n]");
                printf("Please insert a number for this question: ");
                end_taskRead = scanf("%d", &end_task);
            }

                //if-else statement for end_task
                if(end_task==1)
                    {
                        //goto statement and the user can re-enter the input without end the program
                        goto again;
                    }
                else if(end_task==2)
                    {
                        printf("\n=======================END OF PROGRAM==========================\n");
                        break;
                        //end of program
                    }
                else
                    {
                        printf("\n-- Invalid Number !! Please Try Again :) --\n");
                    }
        }
        while(end_task<1 || end_task>2);

}
```

Figure 14: *End_task query for user.*

Arif declared int end_task = 1 to receive input data from the user and int end_taskRead = 0 to invalid an alphabet input from the user. If the condition is end_taskRead !=1 is obeyed, it will loop until the user inserts the input of a number. The %*[^\n] function is to scan everything until a newline \n (does not scan \n) and discards it, that is does not store the scanned data anywhere. If-else statement is used to query the user which they want to end the program or re-enter the input that the program will show the header again. If the user input number 1, the program will go to again: label that the user can re-enter the input without ending the program. For input number 2, the program will end and the user will get a final output END OF PROGRAM. Thus, if the user inserts 0 for end_task, it will loop until the user has the valid input. The valid range for end_task is from 1 to 2.

5.0 OUTPUT DESCRIPTION

Enter a few things here, such as user name, age, weight and height. Users should also enter their gender and level of activity. It is important for users to put in the right information, because if they don't, the programme will keep looping until they enter the correct input. Afterwards, the program will be using their information to figure out BMR and DCR based on that information. Then, all of this information will show up in the output section. The figures below show the outputs when the program is running.

5.1) Output for Male

```
 ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
<                                                                >
<       BMR & DCR CALCULATOR by ALPHA QUATTUOR :v                 >
<                                                                >
 ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Enter Your Name (e.g : Aliff): Arif
-- Your Name is Valid :v --
Enter Your Age (e.g : 20 years): 19
-- Your Age is Valid :v --
Enter Your Weight in kilogram (e.g : 70kg): 75
-- Your Weight is Valid :v --
Enter Your Height in centimeter (e.g: 170cm): 175
-- Your Height is Valid :v --
Enter Your Gender (e.g : 1 : Male or 2 : Female): 1
-- Your Gender is Valid :v --
Enter Your Level of Activity:
1. Sedentary : little or no exercise, desk job
2. Light Activity : light exercise or sports 1-3 days/week
3. Moderate Activity : moderate exercise or sports 3-5 days/week
4. Very Active : hard exercise or sports 6-7 days/week
5. Extra Active : hard daily exercise or sports & physical job or 2x day training, i.e. marathon, contest
Enter your choice: 3
-- Your Level of Activity is Valid :v --


----------------------------Summary-----------------------------
Name    : Arif
Age     : 19 years
Weight  : 75 kg
Height  : 175 cm
Gender  : Male
Level of Activity : Moderate Activity
BMR     : Calculation = 66 + (13.7 x  75 kg) + (5 x  175 cm) - (6.8 x  19 years)
BMR     : 1839 calories
DCR     : Calculation = 1839 calories x 1.55
DCR     : 2850 calories
Note    : Based on this calculation, you would need 2850 calories
          food to sustain your daily activities.

=========================THANK YOU==============================
```

```


Press '1' if you want to re-enter the input and
Press '2' to end this program :) : 2

=======================END OF PROGRAM============================


...Program finished with exit code 0
Press ENTER to exit console.
```

Figure 15: *Output for male user.*

5.2) Output for Female

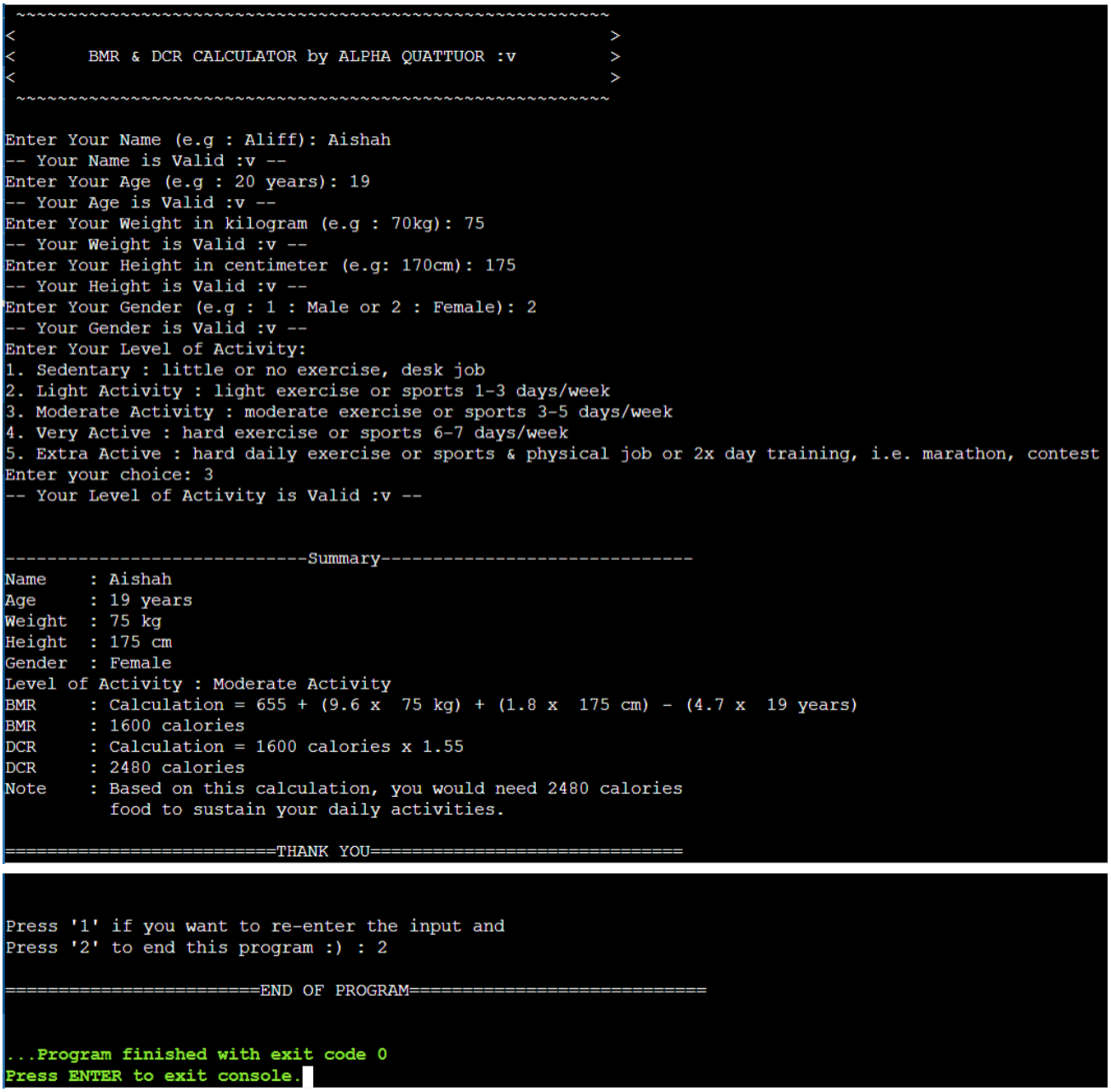


Figure 16: *Output for female user.*

5.3) Output for the user that insert wrong input and re-enter the input

```
 ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
<                                                                    >
<       BMR & DCR CALCULATOR by ALPHA QUATTUOR :v                     >
<                                                                    >
 ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Enter Your Name (e.g : Aliff): Aqmal
-- Your Name is Valid :v --
Enter Your Age (e.g : 20 years): mineteen
--That is not a number!! Please Try Again :) --
Please insert a number for Age : 19
-- Your Age is Valid :v --
Enter Your Weight in kilogram (e.g : 70kg): 19
-- Invalid Weight !! Please Try Again :) --
Enter Your Weight in kilogram (e.g : 70kg): 85
-- Your Weight is Valid :v --
Enter Your Height in centimeter (e.g: 170cm): one six seven
--That is not a number!! Please Try Again :) --
Please insert a number for Height : 168
-- Your Height is Valid :v --
Enter Your Gender (e.g : 1 : Male or 2 : Female): 3
-- Invalid Gender !! Please Try Again :) --
Enter Your Gender (e.g : 1 : Male or 2 : Female): 1
-- Your Gender is Valid :v --
Enter Your Level of Activity:
1. Sedentary : little or no exercise, desk job
2. Light Activity : light exercise or sports 1-3 days/week
3. Moderate Activity : moderate exercise or sports 3-5 days/week
4. Very Active : hard exercise or sports 6-7 days/week
5. Extra Active : hard daily exercise or sports & physical job or 2x day training, i.e. marathon, contest
Enter your choice: two
--That is not a number!! Please Try Again :) --
Please insert a number for Level of Activity : 2
-- Your Level of Activity is Valid :v --
```

```
-----------------------------Summary-----------------------------
Name    : Aqmal
Age     : 19 years
Weight  : 85 kg
Height  : 168 cm
Gender  : Male
Level of Activity : Light Activity
BMR     : Calculation = 66 + (13.7 x  85 kg) + (5 x  168 cm) - (6.8 x  19 years)
BMR     : 1941 calories
DCR     : Calculation = 1941 calories x 1.375
DCR     : 2668 calories
Note    : Based on this calculation, you would need 2668 calories
          food to sustain your daily activities.

==========================THANK YOU==============================


Press '1' if you want to re-enter the input and
Press '2' to end this program :) : 1

 ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
<                                                                    >
<       BMR & DCR CALCULATOR by ALPHA QUATTUOR :v                     >
<                                                                    >
 ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Enter Your Name (e.g : Aliff): Alif
-- Your Name is Valid :v --
Enter Your Age (e.g : 20 years): 19
-- Your Age is Valid :v --
Enter Your Weight in kilogram (e.g : 70kg): 70
-- Your Weight is Valid :v --
Enter Your Height in centimeter (e.g: 170cm): 179
```

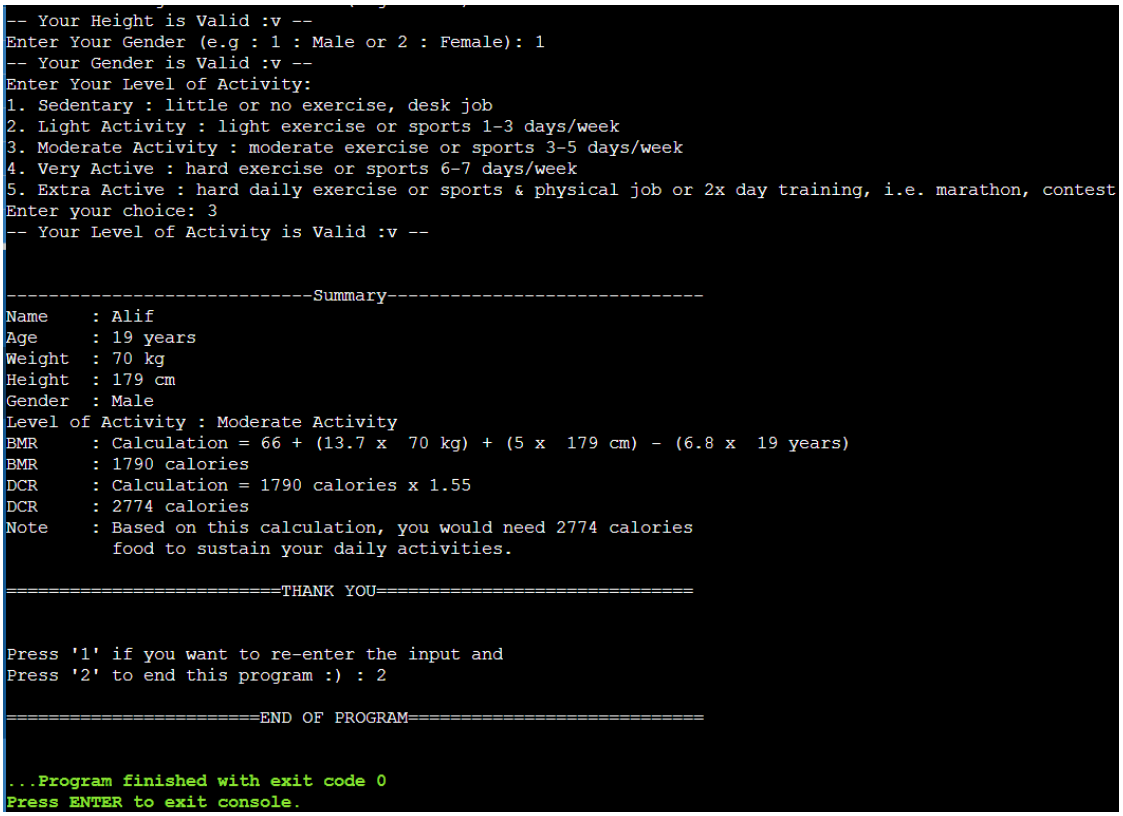


Figure 17: *Output when the wrong data is inserted into the program.*

6.0 DISCUSSION AND CONCLUSION

As a whole, the process of creating this project gave us invaluable experience, as we all worked together to create outstanding programmes that we are all very satisfied with. This is demonstrated by the fact that, based on the coding that we have performed, we have received satisfactory results for the programme in question. We were able to make efficient use of the C programming language in order to discover solutions to the problems that were presented to us. Moreover, we construct a full software that will be extremely beneficial to the community by combining current information with new knowledge obtained from different sources, as well as by integrating it into the existing code to increase the number of functions in this program. We may be willing to help and advise each other in overcoming any obstacles that arise in the process of putting this programme into action. Because each member of the team has their own viewpoint and differing views from the others, it was originally difficult for us all to put this concept into action at first until the group unanimously elected Arif as its leader at the end of the conference. We developed a strong sense of communication inside this team, which indirectly contributed to the development of team bonding as well.

Based on the program that we have done which is a Calories Calculator. Using the Calorie Calculator, people may get an estimate of the amount of calories a person needs to eat on a daily basis. It is also possible to get some easy instructions for gaining or decreasing weight using this calculator.

As a student, there is a great deal more that may be learned in order to become a skilled programmer. With all of the duties that might be daunting at times, we hope that we will be able to develop and accomplish our goals in the most appropriate manner possible. If we live in a programming environment where ideas and creativity are limitless, there is always the possibility that more improvisation will be included in the future. Despite the fact that we will never be able to master everything at once, we will always be able to learn something new every day, and the feeling of achievement that comes from this will serve as our constant source of inspiration to keep trying.

REFERENCES

[1] Paul Deitel and Harvey Deitel, C How to Program with an introduction to C++. 8th ed. New York: USA, 2016.

[2] R. R. Ryan and H. Spiller, The C programming language and a C compiler, in IBM Systems Journal, vol. 24. New York: USA, 1985.

[3] Ye Chen, Ren Zhikao, and Chen Chunping, Research and Development of C Language Programming Experiment Assistant Management Platform Based on Hybrid. Volume 15. Qingdao: China, 2011.

[4] Red Factory, C Programming Notebook: Notebook Of C Programming Code, C Programming. 1st ed. USA , 2019.

[5] M.R. Mousavi and A. De Lucia, Science of Computer Programming. vol. 216. Amsterdam: Netherlands, 2021.

[6] Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language. 2nd ed. New Jersey : USA , 1988.

[7] Bjarne Stroustrup, The C++ Programming Language. 4th ed. New Jersey: USA, 2013.

[8] Scott Meyers, Effective Modern C++ Massachusetts: United States, 2014.

[9] Kyle Loudon, C++ Pocket Reference Massachusetts: United States, 2008.

[10] Michael Dawson, Beginning C++ Through Game Programming. 3rd ed. Florence: United States, 2010.

APPENDIX

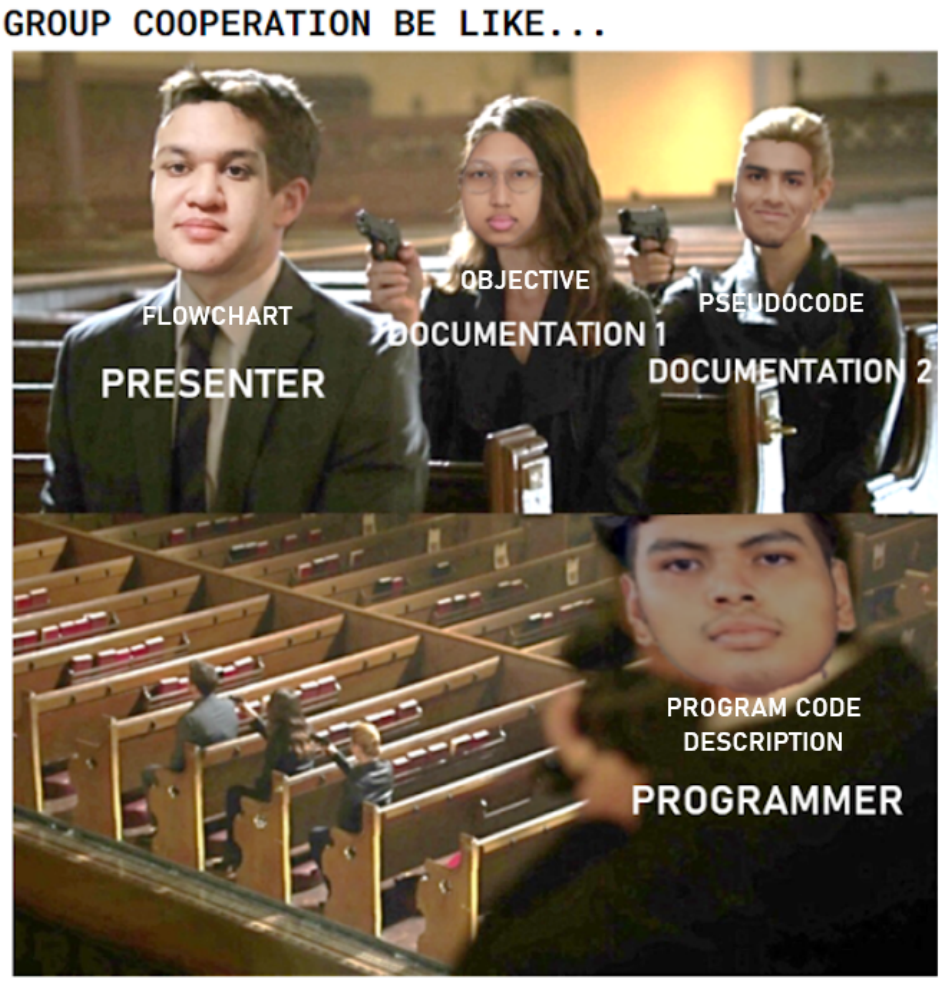


Figure 18: *Memes by Arif.*