UNIVERSITI MALAYSIA SARAWAK

Faculty of Computer Science and Information Technology

Assignment

Assignment Number of Title	Pair Assignment	
Subject Code	TMF 1414	Subject Name: Introduction to Programming

Name	**Matric No**	**Phone No**	**E-mail**
Christopher Sii How Chiong	69385	0196386769	69385@siswa.unimas.my
Joash Paul	69990	0163339721	69990@siswa.unimas.my

Name of Lecturer : Dr. Wang Hui Hui	
Due Date : 3rd November 2019	Date received and approved (for office use only)

Plagiarism and Collusion are methods of cheating that falls under Peraturan Akademik Universiti Malaysia Sarawak para 11: Etika Akademik

Plagiarism

Plagiarism is the presentation of work which has been copied in whole or in part from another person's work, or from any other source such as the internet, published books or periodicals without due acknowledgement given in the text.

Collusion

Collusion is the presentation of work that is the result in whole or in part of unauthorized collaboration with another person or persons.

Where there are reasonable grounds for believing that cheating has occurred, the only action that may be taken when plagiarism or collusion is detected is for the staff member not to mark the item of work and to report or refer the matter to the Dean. This may result in work being disallowed and given a fail grade or if the circumstances warrant, the matter may be referred to a Committee of inquiry for investigation. Such investigation may result in the matter being referred to the University Discipline Committee, which has the power to exclude a student.

Note: Assignments are returned after being assessed by lecturer and to be collected from the lecturer's office. Students should endeavour to collect their assignments prior to commencement of next semester. All unclaimed assignments become the property of the faculty and will become subject to destruction.

Student's Statement :

I certify that I have not plagiarized the work of others or participated in unauthorized collusion when preparing this assignment. I also certify that I have taken proper case in safeguarding my work and have made all reasonable efforts to ensure that may work not be able to be copied.

Signature : ....

Signature : ....Joash..........................

MARK :	Comments:

This cover sheet must be completed, signed and firmly attached to the front of the submission.

All work must be submitted by the due date. If an extension of work is granted, an assignment extension acknowledge slip must be signed by lecturer/ tutor and attached to the assignment.

Please note that it is the student's responsibility to retain a copy of his/her own assignment.

Table of Content

Introduction

We (Chris & Joash) are working at a telephone company that sells pre-paid reload card. We were instructed by our manager to write a program to estimate the total call charge for every customer. Every phone call is charged as table shown below:

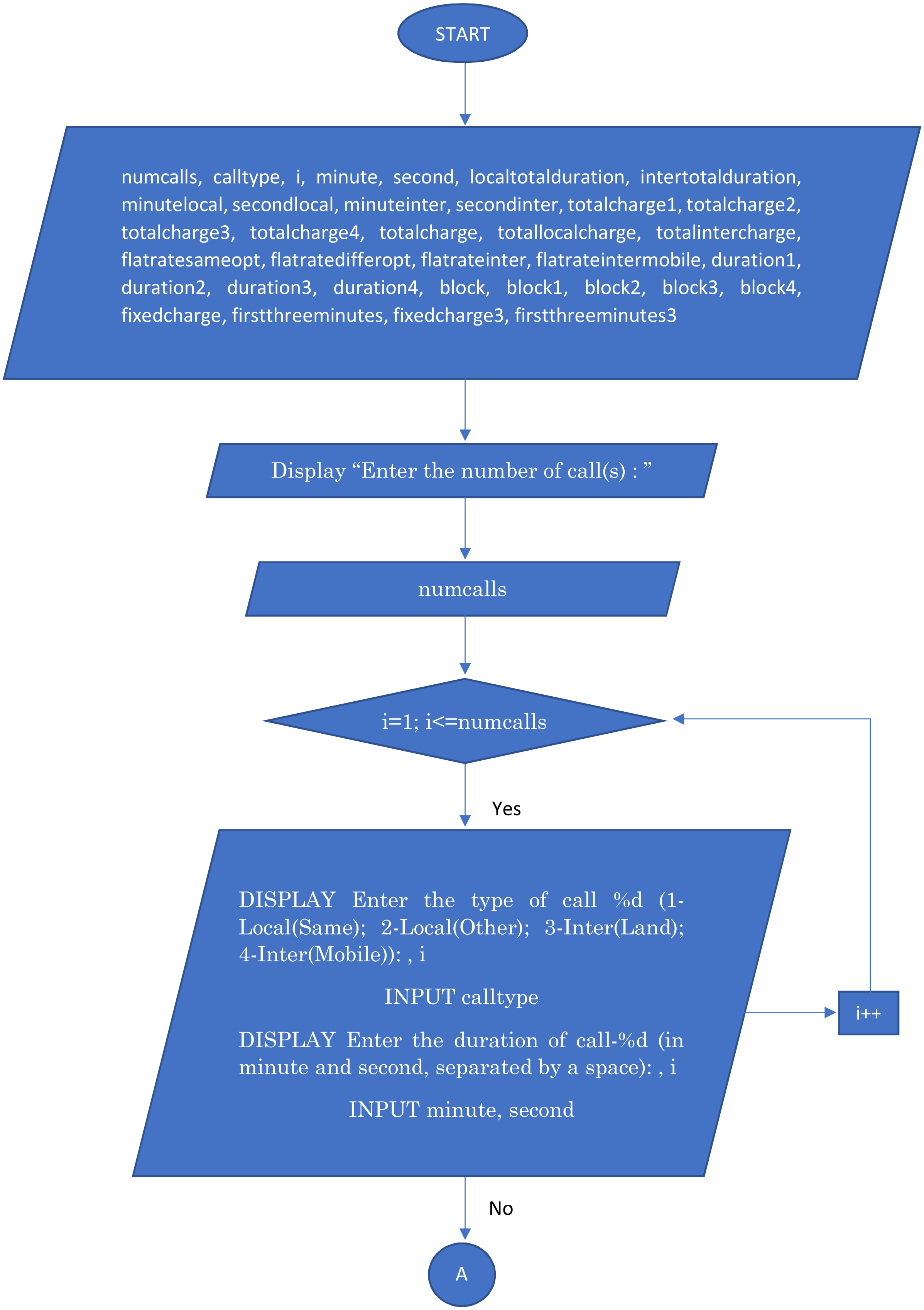
Call Type	**Fixed Charge**	**First 3 minutes**	**Subsequent Minutes**
Local (Same operator)	Flat rate @ RM0.20 / 25 sec block	-	-
Local (Other Operator)	Flat rate @ RM0.35 / 30 sec block	-	-
International (Landline)	RM0.50	RM2.50	RM0.75 / min block
International (Mobile)	RM1.00	RM3.00	RM0.85 / min block

Local call is charged at flat rate in block. A block is a duration between 0 – 25 sec for same operator and 0 – 30 sec for other operator. This means that a call of 1 sec is also considered as one block. The subsequent minutes for international call is in per minute block where a duration less than 1 minute is considered as a block.

Requirements:

- ✓ The program should initially prompt the user for the number of call made.
- ✓ Then input the type and duration for each phone call.
- ✓ Your program should output the total charge for each call, total of duration for local call with total charge as well as duration for international call with total charge. Finally, the total charge for the customer is displayed on the screen.
- ✓ Your program should check for invalid data such as non-positive or number of calls or less than 1, number of second greater than 60 and so on wherever you think necessary. It should prompt the user for corrected input whenever it detects invalid input.
- ✓ Applying appropriate control structures to handle program logic in your program.
  - ➢ Include at least one (1) different type of selection or multiple selection structure
  - ➢ Include at least two (2) different type of repetition structures

Flow charts



numcalls, calltype, i, minute, second, localtotalduration, intertotalduration, minutelocal, secondlocal, minuteinter, secondinter, totalcharge1, totalcharge2, totalcharge3, totalcharge4, totalcharge, totallocalcharge, totalintercharge, flatratesameopt, flatratedifferopt, flatrateinter, flatrateintermobile, duration1, duration2, duration3, duration4, block, block1, block2, block3, block4, fixedcharge, firstthreeminutes, fixedcharge3, firstthreeminutes3

Display "Enter the number of call(s) : "

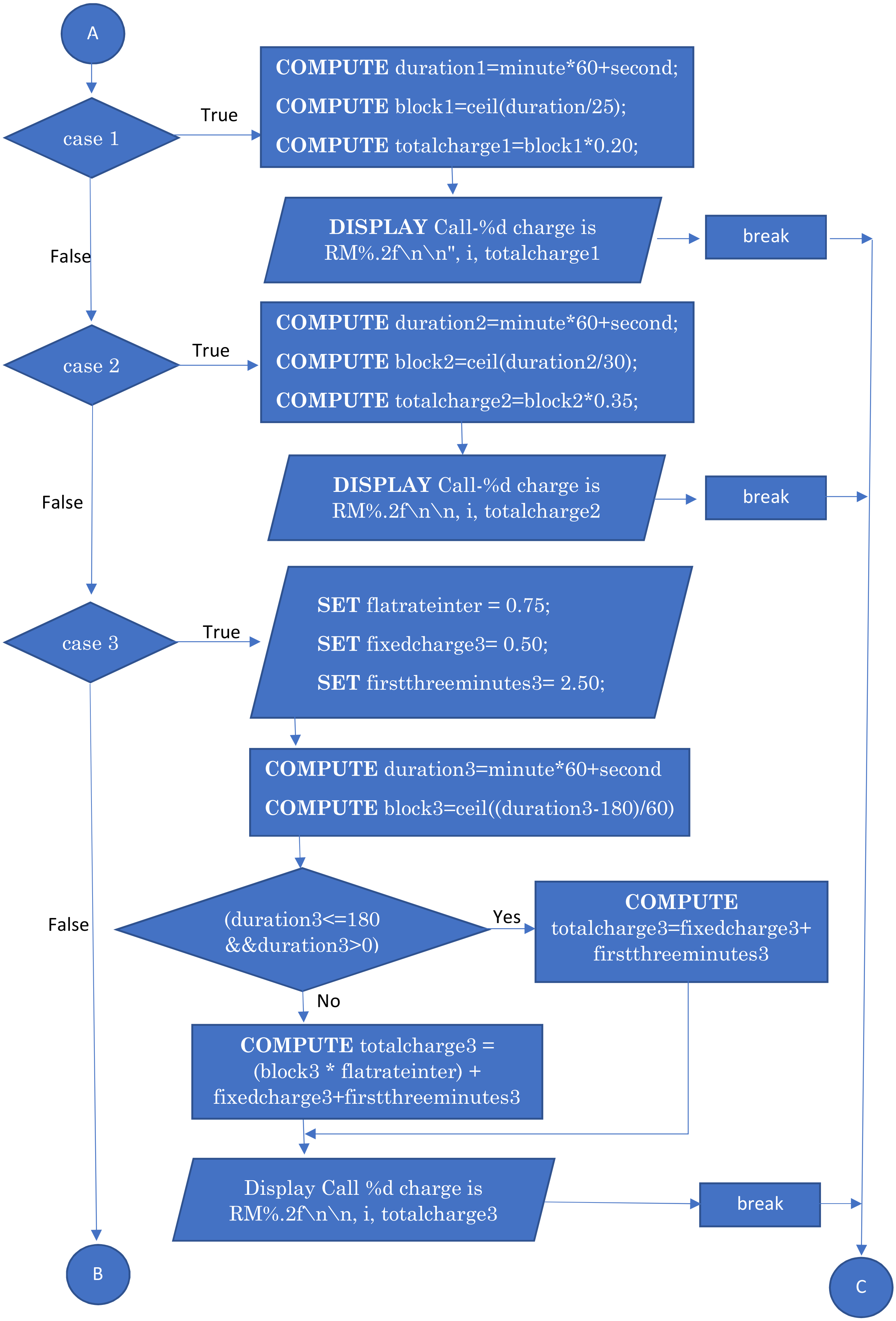
numcalls

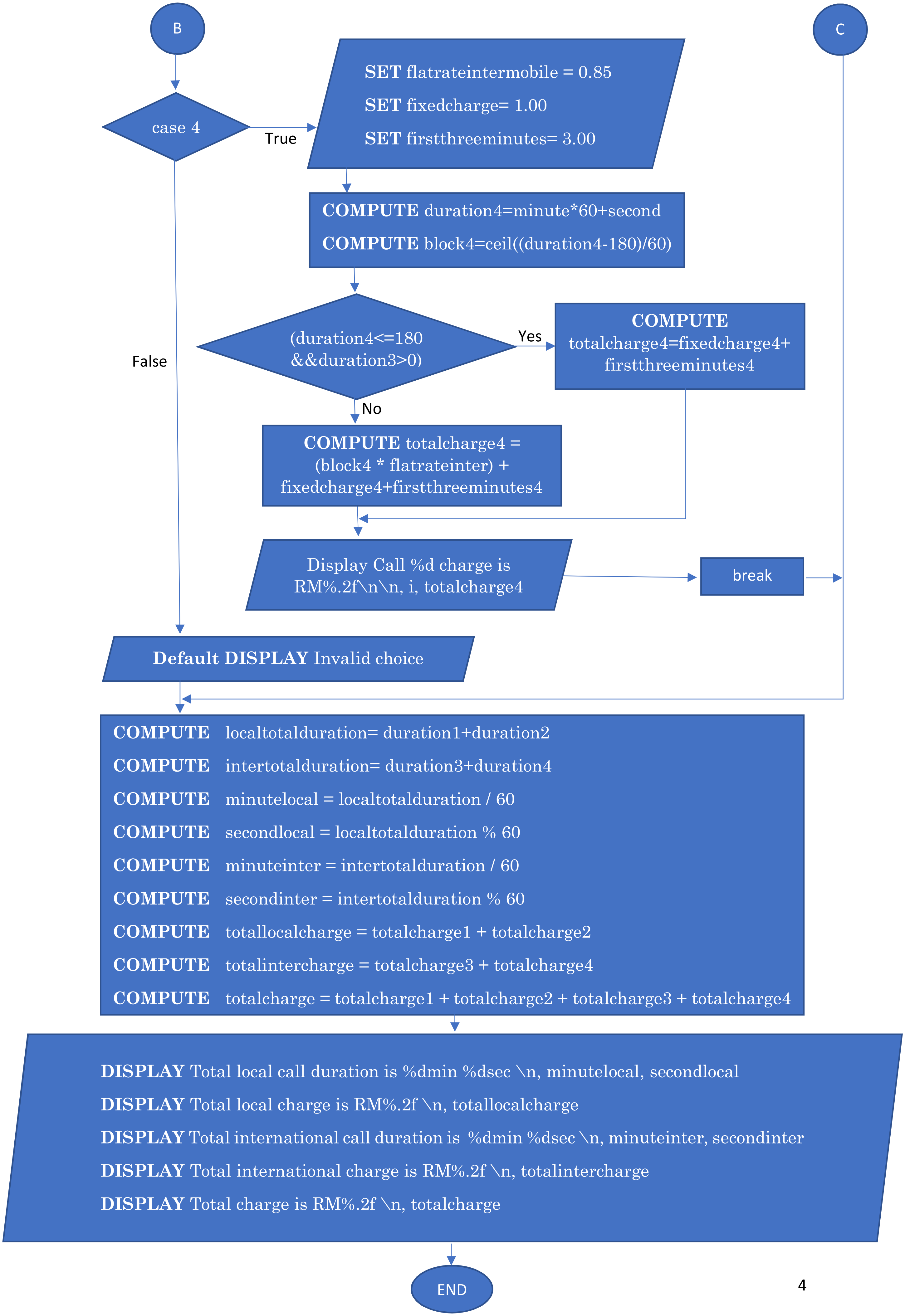
DISPLAY Enter the type of call %d (1-Local(Same); 2-Local(Other); 3-Inter(Land); 4-Inter(Mobile)): , i

INPUT calltype

DISPLAY Enter the duration of call-%d (in minute and second, separated by a space): , i

INPUT minute, second


A
case 1
True
COMPUTE duration1=minute*60+second;
COMPUTE block1=ceil(duration/25);
COMPUTE totalcharge1=block1*0.20;
DISPLAY Call-%d charge is RM%.2f\n\n", i, totalcharge1
break
False
case 2
True
COMPUTE duration2=minute*60+second;
COMPUTE block2=ceil(duration2/30);
COMPUTE totalcharge2=block2*0.35;
DISPLAY Call-%d charge is RM%.2f\n\n, i, totalcharge2
break
False
case 3
True
SET flatrateinter = 0.75;
SET fixedcharge3= 0.50;
SET firstthreeminutes3= 2.50;
COMPUTE duration3=minute*60+second
COMPUTE block3=ceil((duration3-180)/60)
False
(duration3<=180 &&duration3>0)
Yes
COMPUTE totalcharge3=fixedcharge3+ firstthreeminutes3
No
COMPUTE totalcharge3 = (block3 * flatrateinter) + fixedcharge3+firstthreeminutes3
Display Call %d charge is RM%.2f\n\n, i, totalcharge3
break
B
C


B
C
case 4
True
False
SET flatrateintermobile = 0.85
SET fixedcharge= 1.00
SET firstthreeminutes= 3.00
COMPUTE duration4=minute*60+second
COMPUTE block4=ceil((duration4-180)/60)
(duration4<=180 &&duration3>0)
Yes
No
COMPUTE totalcharge4=fixedcharge4+ firstthreeminutes4
COMPUTE totalcharge4 = (block4 * flatrateinter) + fixedcharge4+firstthreeminutes4
Display Call %d charge is RM%.2f\n\n, i, totalcharge4
break
Default DISPLAY Invalid choice
COMPUTE localtotalduration= duration1+duration2
COMPUTE intertotalduration= duration3+duration4
COMPUTE minutelocal = localtotalduration / 60
COMPUTE secondlocal = localtotalduration % 60
COMPUTE minuteinter = intertotalduration / 60
COMPUTE secondinter = intertotalduration % 60
COMPUTE totallocalcharge = totalcharge1 + totalcharge2
COMPUTE totalintercharge = totalcharge3 + totalcharge4
COMPUTE totalcharge = totalcharge1 + totalcharge2 + totalcharge3 + totalcharge4
DISPLAY Total local call duration is %dmin %dsec \n, minutelocal, secondlocal
DISPLAY Total local charge is RM%.2f \n, totallocalcharge
DISPLAY Total international call duration is %dmin %dsec \n, minuteinter, secondinter
DISPLAY Total international charge is RM%.2f \n, totalintercharge
DISPLAY Total charge is RM%.2f \n, totalcharge
END

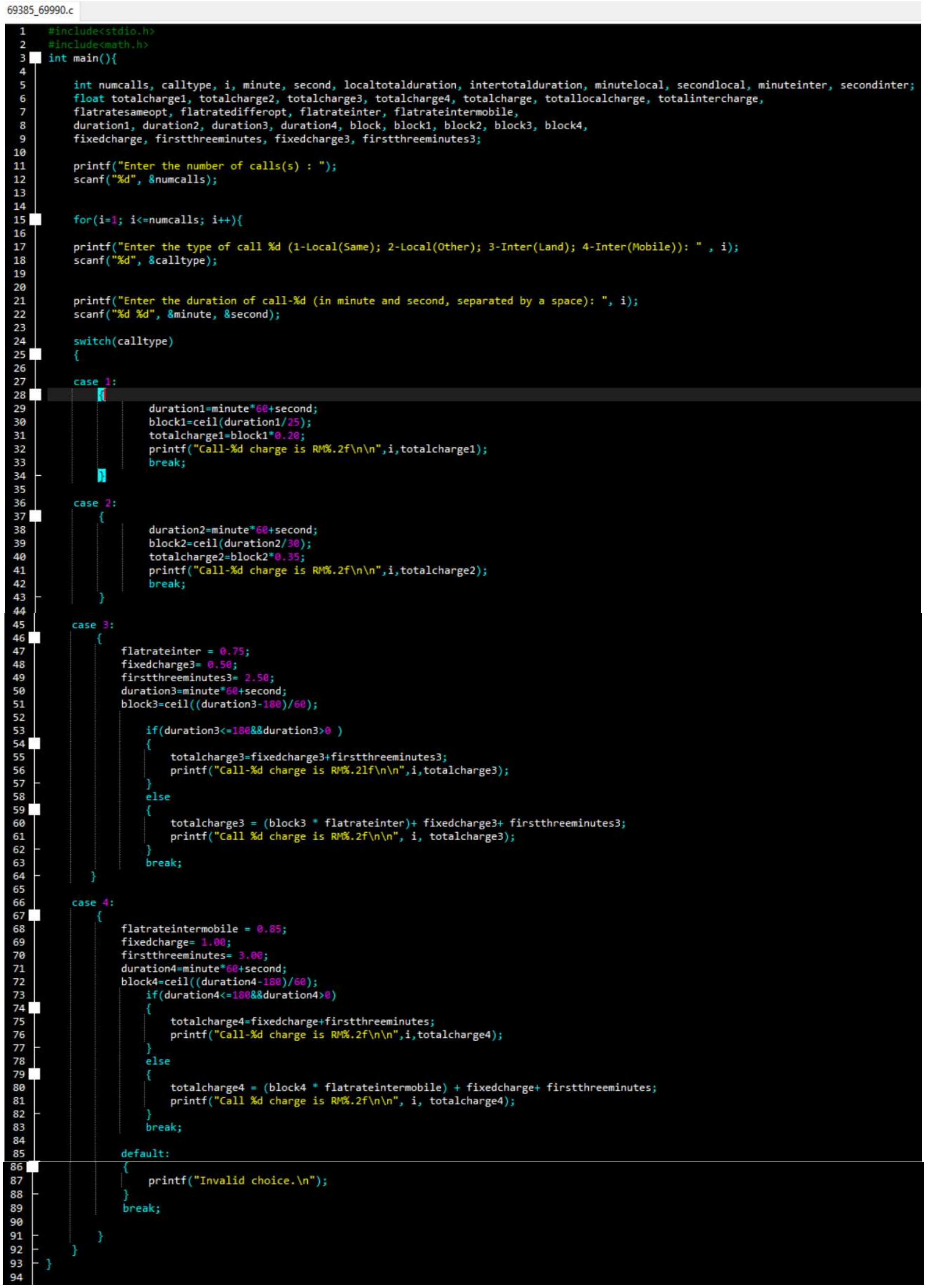
Pseudocode

```
START

GET         numcalls, calltype, i, minute, second, localtotalduration, intertotalduration,
            minutelocal, secondlocal, minuteinter, secondinter, totalcharge1,
            totalcharge2, totalcharge3, totalcharge4, totalcharge, totallocalcharge,
            totalintercharge, flatratesameopt, flatratedifferopt, flatrateinter,
            flatrateintermobile, duration1, duration2, duration3, duration4, block,
            block1, block2, block3, block4, fixedcharge, firstthreeminutes, fixedcharge3,
            firstthreeminutes3;


DISPLAY     “Enter the number of call(s) : ”


INPUT       numcalls;


FOR   (i=1; i<=numcalls; i++)
      DISPLAY     "Enter the type of call %d (1-Local(Same); 2-Local(Other);
                    3-Inter(Land); 4-Inter(Mobile)): ", i;
      INPUT       calltype;
      DISPLAY     "Enter the duration of call-%d (in minute and second, separated by
                    a space): ", i;
      INPUT       minute, second;
ENDFOR


CASE calltype of
      1:    COMPUTE     duration1=minute*60+second;
            COMPUTE     block1=ceil(duration/25);
            COMPUTE     totalcharge1=block1*0.20;
            DISPLAY     “"Call-%d charge is RM%.2f\n\n", i, totalcharge1;

      2:    COMPUTE     duration2=minute*60+second;
            COMPUTE     block2=ceil(duration2/30);
            COMPUTE     totalcharge2=block2*0.35;
            DISPLAY     "Call-%d charge is RM%.2f\n\n", i, totalcharge2;
```

```
        3:  SET         flatrateinter = 0.75;
            SET         fixedcharge3= 0.50;
            SET         firstthreeminutes3= 2.50;
            COMPUTE     duration3=minute*60+second;
            COMPUTE     block3=ceil((duration3-180)/60);

            IF      (duration3<=180&&duration3>0)
                    COMPUTE     totalcharge3=fixedcharge3+firstthreeminutes3;
                    DISPLAY     "Call-%d charge is RM%.2lf\n\n",i,totalcharge3);
            ELSE
                    COMPUTE     totalcharge3 = (block3 * flatrateinter) + fixedcharge3+
                                               firstthreeminutes3;
                    DISPLAY "Call %d charge is RM%.2f\n\n", i, totalcharge3;
            ENDIF

        4:  SET         flatrateintermobile = 0.85;
            SET         fixedcharge= 1.00;
            SET         firstthreeminutes= 3.00;
            COMPUTE     duration4=minute*60+second;
            COMPUTE     block4=ceil((duration4-180)/60);

            IF      (duration4<=180&&duration4>0)
                    COMPUTE     totalcharge4=fixedcharge+firstthreeminutes;
                    DISPLAY     "Call-%d charge is RM%.2f\n\n",i,totalcharge4;
            ELSE
                    COMPUTE     totalcharge4 = (block4 * flatrateintermobile) +
                                               fixedcharge + firstthreeminutes;
                    DISPLAY     "Call %d charge is RM%.2f\n\n", i, totalcharge4;
            ENDIF

    OTHERS
            DISPLAY     "Invalid choice.\n";

    ENDCASE
```

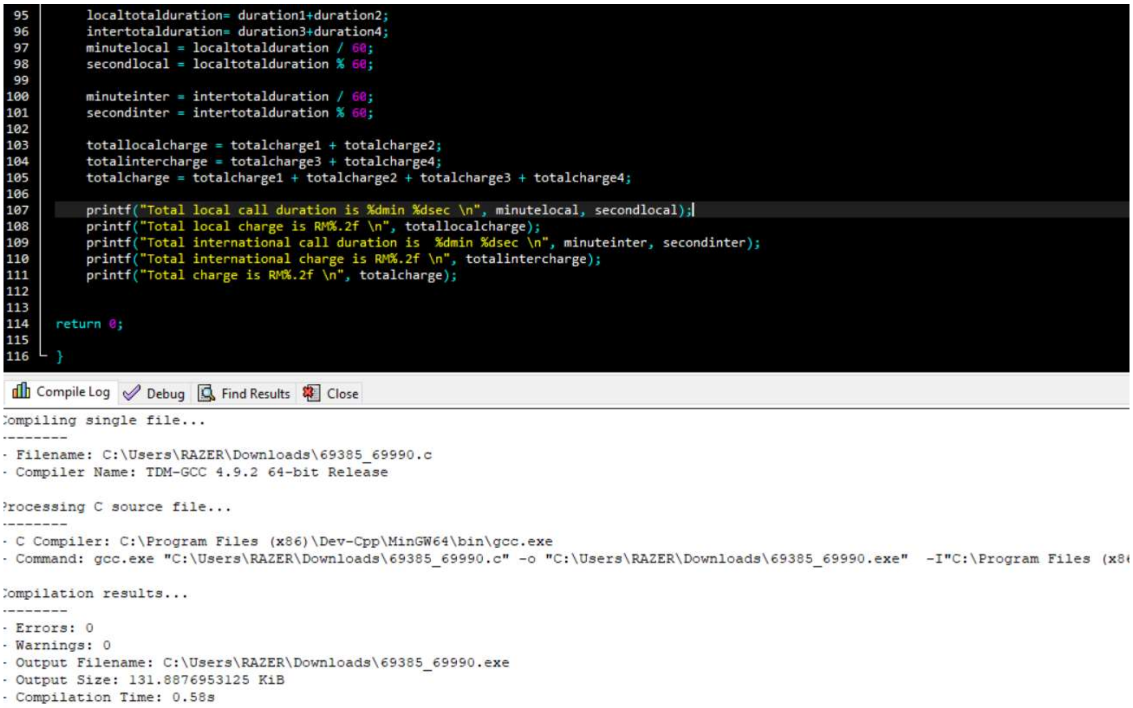
```
COMPUTE   localtotalduration= duration1+duration2;
COMPUTE   intertotalduration= duration3+duration4;
COMPUTE   minutelocal = localtotalduration / 60;
COMPUTE   secondlocal = localtotalduration % 60;
COMPUTE   minuteinter = intertotalduration / 60;
COMPUTE   secondinter = intertotalduration % 60;
COMPUTE   totallocalcharge = totalcharge1 + totalcharge2;
COMPUTE   totalintercharge = totalcharge3 + totalcharge4;
COMPUTE   totalcharge = totalcharge1 + totalcharge2 + totalcharge3 + totalcharge4;

DISPLAY   "Total local call duration is %dmin %dsec \n", minutelocal, secondlocal;
DISPLAY   "Total local charge is RM%.2f \n", totallocalcharge;
DISPLAY   "Total international call duration is  %dmin %dsec \n", minuteinter,
            secondinter;
DISPLAY   "Total international charge is RM%.2f \n", totalintercharge;
DISPLAY   "Total charge is RM%.2f \n", totalcharge;

END
```

Print screens



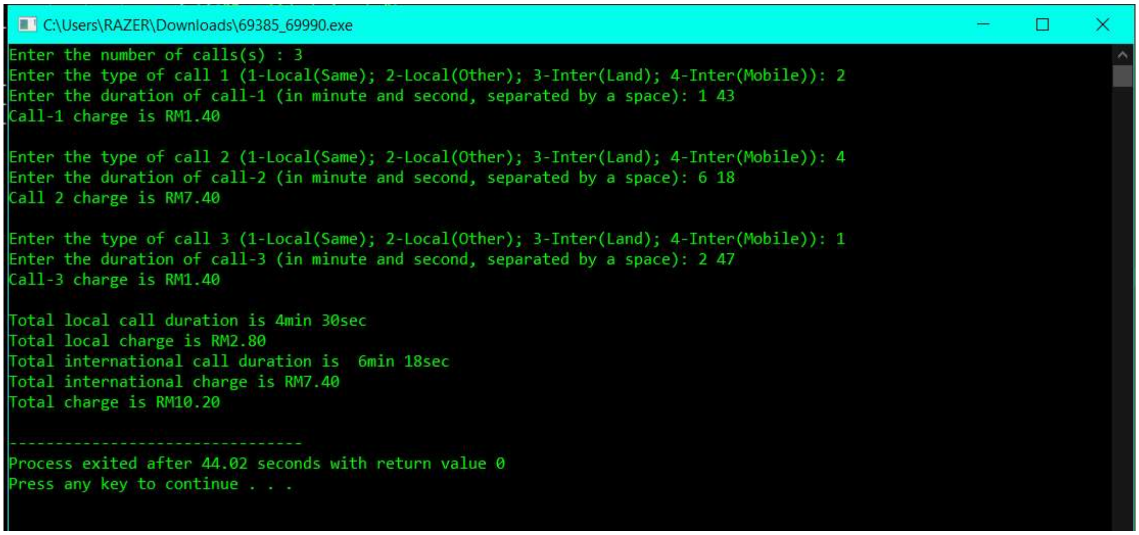
69385_69990.c

```c
#include<stdio.h>
#include<math.h>
int main(){

    int numcalls, calltype, i, minute, second, localtotalduration, intertotalduration, minutelocal, secondlocal, minuteinter, secondinter;
    float totalcharge1, totalcharge2, totalcharge3, totalcharge4, totalcharge, totallocalcharge, totalintercharge,
    flatratesameopt, flatratedifferopt, flatrateinter, flatrateintermobile,
    duration1, duration2, duration3, duration4, block, block1, block2, block3, block4,
    fixedcharge, firstthreeminutes, fixedcharge3, firstthreeminutes3;

    printf("Enter the number of calls(s) : ");
    scanf("%d", &numcalls);


    for(i=1; i<=numcalls; i++){

    printf("Enter the type of call %d (1-Local(Same); 2-Local(Other); 3-Inter(Land); 4-Inter(Mobile)): " , i);
    scanf("%d", &calltype);


    printf("Enter the duration of call-%d (in minute and second, separated by a space): ", i);
    scanf("%d %d", &minute, &second);

    switch(calltype)
    {

    case 1:
        {
                duration1=minute*60+second;
                block1=ceil(duration1/25);
                totalcharge1=block1*0.20;
                printf("Call-%d charge is RM%.2f\n\n",i,totalcharge1);
                break;
        }

    case 2:
        {
                duration2=minute*60+second;
                block2=ceil(duration2/30);
                totalcharge2=block2*0.35;
                printf("Call-%d charge is RM%.2f\n\n",i,totalcharge2);
                break;
        }

    case 3:
        {
            flatrateinter = 0.75;
            fixedcharge3= 0.50;
            firstthreeminutes3= 2.50;
            duration3=minute*60+second;
            block3=ceil((duration3-180)/60);

                if(duration3<=180&&duration3>0 )
                {
                    totalcharge3=fixedcharge3+firstthreeminutes3;
                    printf("Call-%d charge is RM%.2lf\n\n",i,totalcharge3);
                }
                else
                {
                    totalcharge3 = (block3 * flatrateinter)+ fixedcharge3+ firstthreeminutes3;
                    printf("Call %d charge is RM%.2f\n\n", i, totalcharge3);
                }
                break;
        }

    case 4:
        {
            flatrateintermobile = 0.85;
            fixedcharge= 1.00;
            firstthreeminutes= 3.00;
            duration4=minute*60+second;
            block4=ceil((duration4-180)/60);
                if(duration4<=180&&duration4>0)
                {
                    totalcharge4=fixedcharge+firstthreeminutes;
                    printf("Call-%d charge is RM%.2f\n\n",i,totalcharge4);
                }
                else
                {
                    totalcharge4 = (block4 * flatrateintermobile) + fixedcharge+ firstthreeminutes;
                    printf("Call %d charge is RM%.2f\n\n", i, totalcharge4);
                }
                break;

            default:
            {
                printf("Invalid choice.\n");
            }
            break;

        }
    }
}

```



```c
    localtotalduration= duration1+duration2;
    intertotalduration= duration3+duration4;
    minutelocal = localtotalduration / 60;
    secondlocal = localtotalduration % 60;

    minuteinter = intertotalduration / 60;
    secondinter = intertotalduration % 60;

    totallocalcharge = totalcharge1 + totalcharge2;
    totalintercharge = totalcharge3 + totalcharge4;
    totalcharge = totalcharge1 + totalcharge2 + totalcharge3 + totalcharge4;

    printf("Total local call duration is %dmin %dsec \n", minutelocal, secondlocal);
    printf("Total local charge is RM%.2f \n", totallocalcharge);
    printf("Total international call duration is  %dmin %dsec \n", minuteinter, secondinter);
    printf("Total international charge is RM%.2f \n", totalintercharge);
    printf("Total charge is RM%.2f \n", totalcharge);


return 0;

}
```

```
Compiling single file...
--------
- Filename: C:\Users\RAZER\Downloads\69385_69990.c
- Compiler Name: TDM-GCC 4.9.2 64-bit Release

Processing C source file...
--------
- C Compiler: C:\Program Files (x86)\Dev-Cpp\MinGW64\bin\gcc.exe
- Command: gcc.exe "C:\Users\RAZER\Downloads\69385_69990.c" -o "C:\Users\RAZER\Downloads\69385_69990.exe"  -I"C:\Program Files (x8

Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\RAZER\Downloads\69385_69990.exe
- Output Size: 131.8876953125 KiB
- Compilation Time: 0.58s
```



```
Enter the number of calls(s) : 3
Enter the type of call 1 (1-Local(Same); 2-Local(Other); 3-Inter(Land); 4-Inter(Mobile)): 2
Enter the duration of call-1 (in minute and second, separated by a space): 1 43
Call-1 charge is RM1.40

Enter the type of call 2 (1-Local(Same); 2-Local(Other); 3-Inter(Land); 4-Inter(Mobile)): 4
Enter the duration of call-2 (in minute and second, separated by a space): 6 18
Call 2 charge is RM7.40

Enter the type of call 3 (1-Local(Same); 2-Local(Other); 3-Inter(Land); 4-Inter(Mobile)): 1
Enter the duration of call-3 (in minute and second, separated by a space): 2 47
Call-3 charge is RM1.40

Total local call duration is 4min 30sec
Total local charge is RM2.80
Total international call duration is  6min 18sec
Total international charge is RM7.40
Total charge is RM10.20

--------------------------------
Process exited after 44.02 seconds with return value 0
Press any key to continue . . .
```