CONFIDENTIAL

UTM Fakulti Komputeran
UNIVERSITI TEKNOLOGI MALAYSIA

76.5
+2
78.5
+1.5
80

UNIVERSITI TEKNOLOGI MALAYSIA
TEST 1 (WRITTEN)
SEMESTER 1, 2023/2024

SUBJECT CODE	:	**SECJ1013**
SUBJECT NAME	:	**PROGRAMMING TECHNIQUE I**
TIME	:	**2 HOURS (8:00PM - 10:00PM)**
DATE/DAY	:	**05 DECEMBER 2023 (TUESDAY)**
VENUES	:	**N28, BK1-7**

INSTRUCTIONS:

This test book consists of two parts:

PART A: PROBLEM-SOLVING (4 QUESTIONS)	(40 MARKS)
PART B: PROGRAMMING (4 QUESTIONS)	(60 MARKS)
TOTAL	**(100 MARKS)**

ANSWER ALL QUESTIONS IN THIS BOOKLET IN THE SPACES PROVIDED.

Additional answer sheets will be given upon request.

(Please Write Your Name, Lecturer Name and Section in Your Answer Booklet)

Name	Lau Jee Wen
I/C No.	041214040018
Year / Course	2023 / SECPH
Section	03
Lecturer Name	Dr. Nies

This question paper consists of **18** printed pages excluding this page.

PART A: PROBLEM-SOLVING QUESTIONS [40 MARKS]

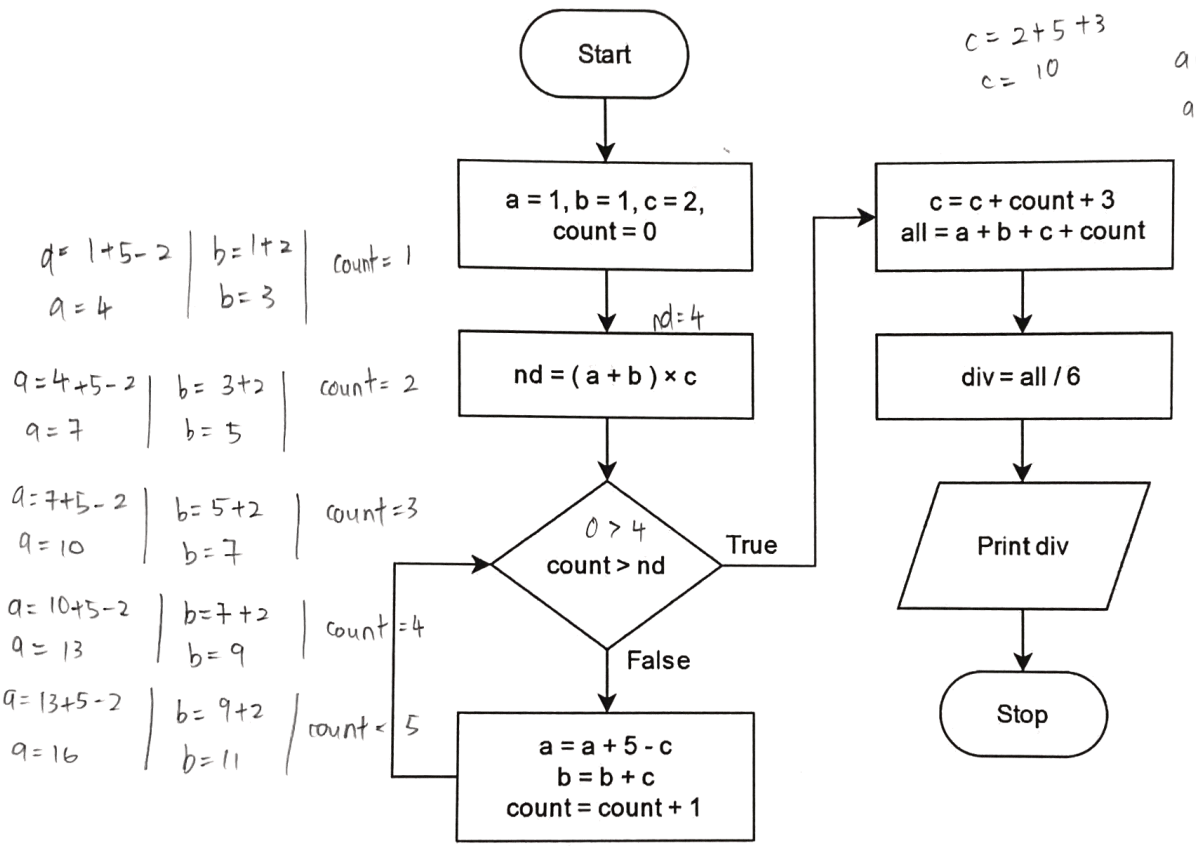
QUESTION 1 [10 MARKS]

Identify the output of the flowchart in **Figure A1** by tracing the value of the variables. Complete the tracing table in **Table A1** to write your answers. Then, show the calculations for variable ***nd***, ***all*** and ***div*** in **Table A2**.

nd = (1+1) × 2
nd = 4

c = 2+5+3
c = 10

all = 16 + 11 + 10 + 5
all = ~~24~~ 42



nd=4

a = 1+5-2 | b = 1+2 | count = 1
a = 4 | b = 3

a = 4+5-2 | b = 3+2 | count = 2
a = 7 | b = 5

a = 7+5-2 | b = 5+2 | count = 3
a = 10 | b = 7

a = 10+5-2 | b = 7+2 | count = 4
a = 13 | b = 9

a = 13+5-2 | b = 9+2 | count < 5
a = 16 | b = 11

div = ~~24~~ 42/6
div = 7

Figure A1: Flowchart for Question 1

Table A1: Tracing table

4

a	b	c	count	count > nd	Output
1	*1*	*2*	*0*	False	–
4	3	2	1	False	–
7	5	2	2	False	–
10	7	2	3	False	–
13	9	2	4	False	–
16	11	2	5	True	–
16	11	10	5	True	7

Table A2: Variables calculation for Question 1

Calculation for variable *nd*, *all* and *div*:

nd = (1+1)×2
nd = 4

all = 16+11+10+5
all = 42

div = 42/6
= 7

10

QUESTION 2 **[10 MARKS]**

Draw a **flowchart** to calculate the price of items after discount. The steps for calculating are as follows:

(a) Get the prices of items. > 50

(b) If the total prices of the items are more than RM 50.00, call a user-defined function named *"disc10"* then calculate the total price after discount and return it to the main function.

disc15(x)
↓
price_dis = x * 1.1
↓
return price_dis

(c) If the total prices of the items are more than RM 100.00, call a user-defined function name *"disc15"* then calculate the total price after discount and return it to the main function.

(d) Finally, in the main function, display the total number of items, the original total price before discount and the price after discount. 总数

set tot_price = 0
set num = 0
while (num >= 0)
{
read price
tot_price = tot_price + price
num ++
}
if (tot_price > 50)
Display num

Table A3 shows the discount tag offer for the total price of the items.

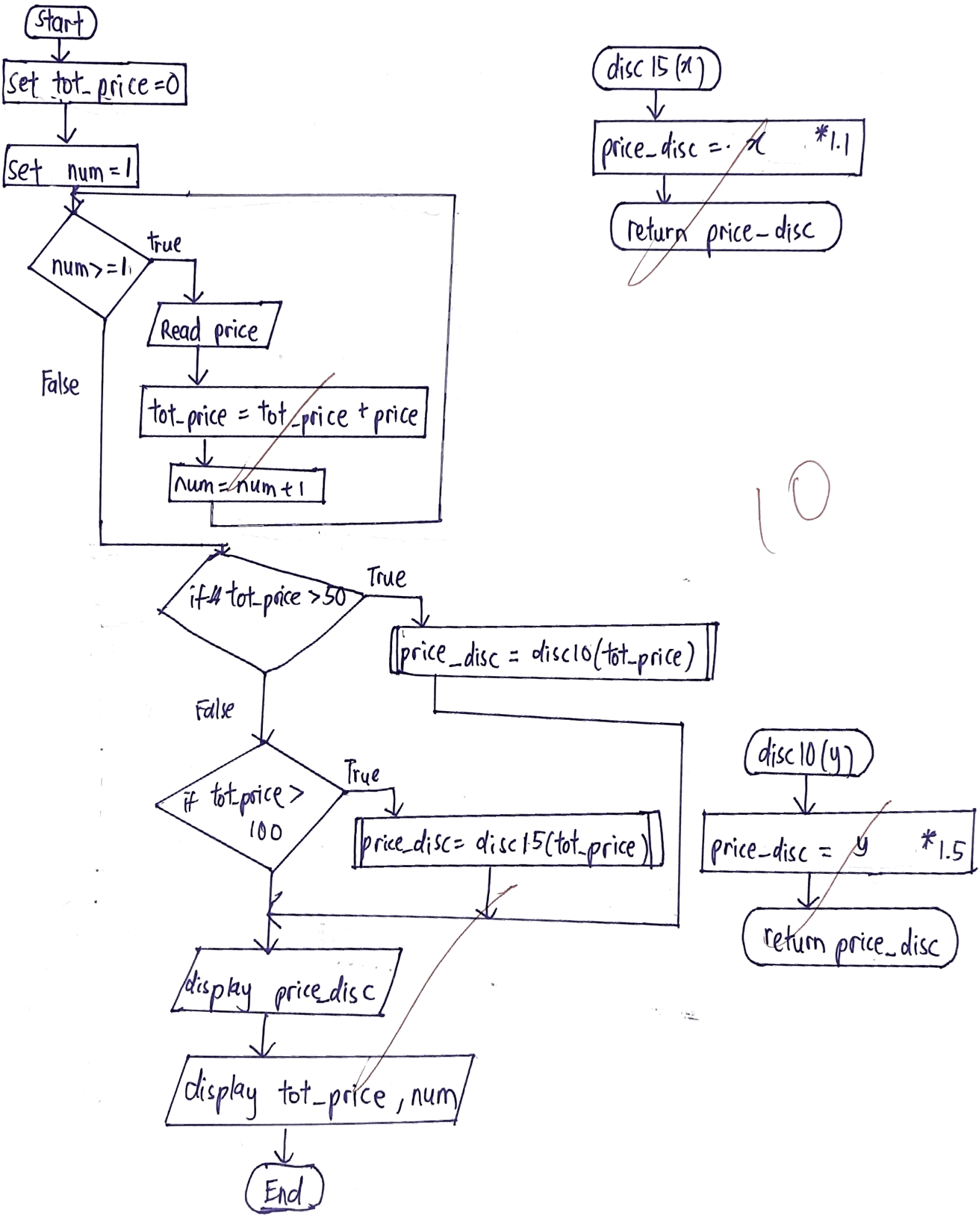
Table A3: Discount for each tag for Question 2

Discount Tag	Discount
disc10	10%
disc15	15%

disc10(x)
↓
price_dis = 1
↓

Note: User input in step (a), allows more than an item price to be added. Calculation of the total price after discount must be done in the user defined function as named in (b) and (c) accordingly. The price after discount must be returned to the main function once the calculation is finished.

Answer space for Question 2



10

QUESTION 3 **[10 MARKS]**

Please complete the flow chart given in **Figure A2** so that it will produce any outputs given in the **example 1 to 3**. Write your answer in **Table A4**.

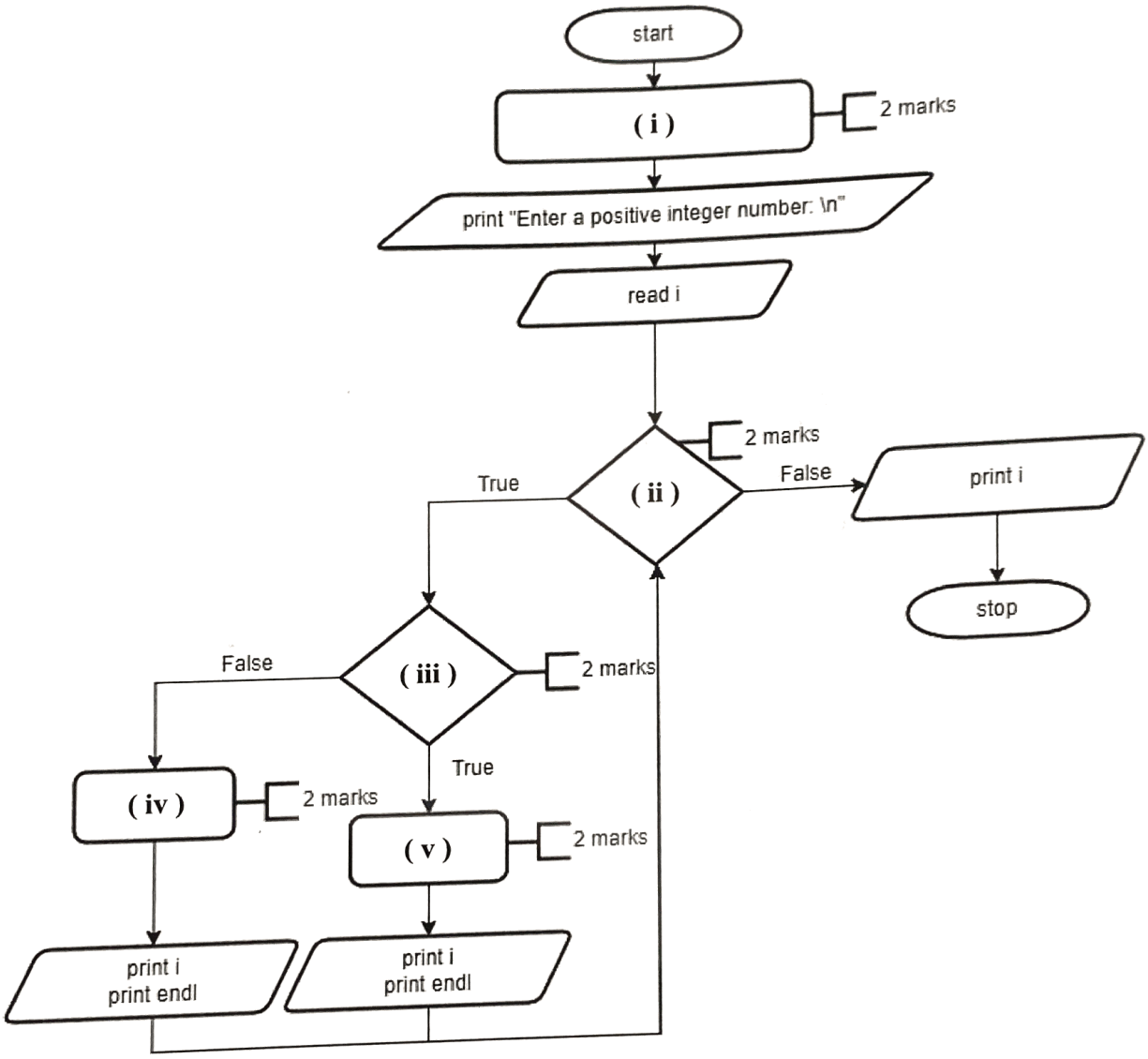


Figure A2: Flowchart for Question 3

Example output 1:

```
Enter a positive integer
number:
23 [Enter]
23
```

Example output 2:

```
Enter a positive integer
number:
14 [Enter]
15
15
```

Example output 3:

```
Enter a positive integer
number:
9 [Enter]
11
11
12
13
14
15
15
```

+2

+1

Note: the bolded text are the inputs from the user

Table A4: Answer for Question 3

Question	Answer
(i)	if(i%3 == 1)
(ii)	while (
(iii)	while (
(iv)	i=i+1
(v)	i=i+2

9 x2

QUESTION 4 **[10 MARKS]**

Write a complete **pseudocode** which will get several inputs from the user: bank 'account balance', the 'account type' and 'account level' they have. Apply **selection** concept. Based on this information and the rate table from Bank ABC in **Table A5**, determine the interest rate they are receiving.

>

Table A5: Rate table of Bank ABC for Question 4

Type of Account	Level	Minimum Balance	Interest Rate
Student	Standard	RM25	1.3 %
Personal	Standard	RM0	1.2 %
Personal	Gold	RM1000	1.9 %
Personal	Gold	RM5000	2.3 %
Business	Standard	RM1500	1.7 %
Business	Platinum	RM10000	2.5%

Answer space for Question 4

1. Start
2. Read balance
3. Read type
4. Read level
5. if(type == "Student")
   5.1 if (balance >= 25)
      5.1.1 interest = 0.013
   5.2 end if
6. else if (type == "Personal")
   6.1 if(level == "Standard" && balance >= 0)
      6.1.1 interest = 0.012
   end if
   6.2 if(level == " Gold ")
      6.2.1 if (balance >= 1000 && balance <5000)
         6.2.1.1 interest = 0.019
      6.2.2 else
         6.2.2.1 interest = 0.023
      6.2.3 end if
   6.3 end if
7. else if (type == " Business")
   7.1 if(level == "Standard" && balance >= 1500)
      7.1.1 interest = 0.017
   7.2 else if (level == "Platinum" && balance >= 10000)
      7.2.2 interest = 0.025
   7.3 end if
8. end if
9. end

10

PART B: PROGRAMMING QUESTIONS [60 MARKS]

QUESTION 1 [15 MARKS]

(a) **Figure B1** shows the flowchart that asks the user to calculate the total mark of a test. The C++ code of the flowchart is shown in **Figure B2**. You are required to complete the C++ code in **Figure B2** and write your answer in **Table B1**. [5 marks]

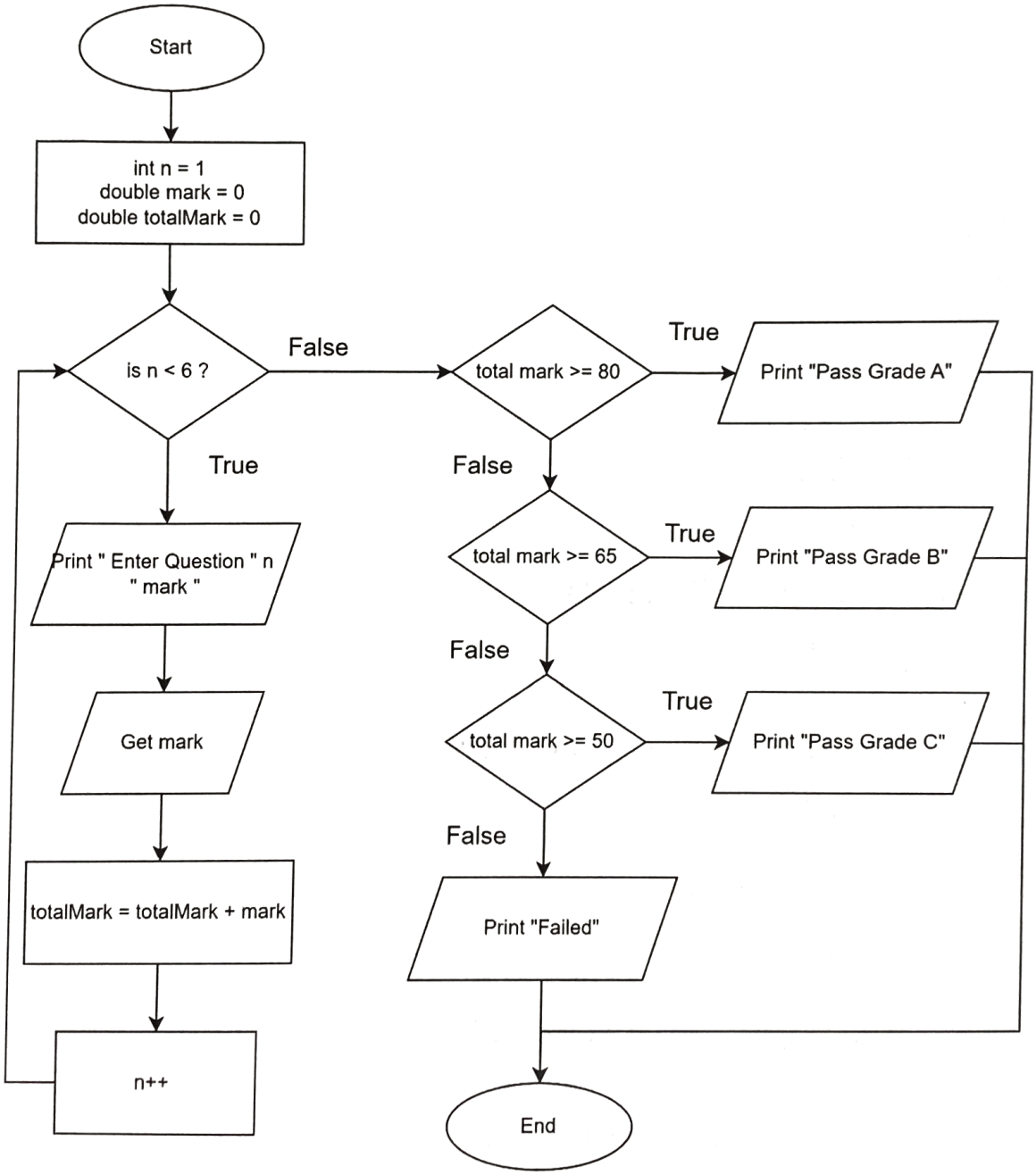


Figure B1: Flowchart for Question 1(a)

```
#include <iostream>
using namespace std;
int main() {

    int n = 1;
    double mark, totalMark = 0;

    while (n <6)                                  (i)
    {
        cout << "Enter Question"<< n <<" mark ";
        cin >> mark;

        totalMark = totalMark + mark ;            (ii)

        n++;
    }


    if(totalMark >= 80)                           (iii)_
        cout << "Pass Grade A";

    if(total Mark >= 65)                          (iv)_____
        cout << "Pass Grade B";

    if(total Mark > = 50)                         (v)_____
        cout << "Pass Grade C";

    else
        cout << "Failed";

    return 0;
}
```

Figure B2: C++ program code for Question 1(a)

Table B1: Answer for Question 1(a)

Question	Answer
(i)	while (n<6)
(ii)	totalMark = totalMark + mark ;
(iii)	if (totalMark >= 80)
(iv)	if (totalMark >= 65)
(v)	if (totalMark >= 50)

(b) From the C++ program code in **Figure B3**, trace the output and write your answer in the answer column. [5 marks]

Line	Code	Answer
1	`#include <iostream>`	
2	`using namespace std;`	
3	`int main(){`	
4	`int a, b = 2, c, d = 1, e=0;`	
5	`float p = 3.0, q=0.0;`	
6	`char x='1', y='A', z='?';`	
7	`bool m=0;`	
8	`if (b>m)` 2>0 (T)	
9	`m=b;` m= 2 m= 1	
10	`else`	
11	`m=d;`	
12		
13	`if (b>e)` 2>0 (T)	
14	`e=e+2;` e=0+2 e=2	
15	`else`	
16	`a=b;` q= (2/5) * 1 =0.4	
17	`q = static_cast<float>(e/5) * d;`	
18	`p = static_cast <float>(b*d--);`	
19	`c = ++a / 2 + e;` p=(2*1)	
20	c= p=2.0	
21	`if (x==1)`	
22	`m=1;` m=1	
23	`else`	
24	`y='D';`	
26	`cout << a << endl;`	1
27	`cout << b << endl;`	2
28	`cout << p << endl;`	2.0
29	`cout << q << endl;`	0.4 X 0
30	`cout << y << endl;`	'A' X D
31	`return 0;`	
32	`}`	

Figure B3: C++ program code for Question 1(b)

2

(c) Identify the output for the program segment shown in **Table B2**. Fill in the **Output** column of the **Table B2** with your answers. [5 marks]

-3 != 0 (✓)
-3 > 0 (x)
-3 < 0 (✓)
0 != 0 (x)
0 > 0 (x)
0 < 0 (x)

Table B2: Program segment and output for Question 1(c)

Line	Code	Output
1	`int x = 10, y = -3;`	
2	`bool k = true;`	
3		
4	`cout << ( (x == 10) || (x < y) ) << endl;` (T) || 10 < -3 (F)	1
5	`cout << ( (x = y) && (k = 0) ) << endl;` x = -3 && k = 0 (F)	0
6	`cout << ( (y == -3) && (x == 10) ) << endl;` (T) (F)	0
7	`cout << ( (y != x) || (k != true) ) << endl;` -3 != -3 (F) || (T)	~~0~~ 1
8	`cout << ( (k == true) && (x >= 10) ) << endl;` (F) -3 >= 10 (F)	0

4

QUESTION 2 **[15 MARKS]**

Figure B4 below shows the complete C++ code.

```
1   #include <iostream>
2   using namespace std;
3
4   int main()
5   {                 char
6      float choice;
7      int counter;
8      bool flag = 0;
9
10     cout << "\tEnter choice    :  ";
11     cin >> choice;
12     cout << "\tEnter counter   :  ";
13     cin >> counter;
14
15     while (!flag && (counter % 2==0))
16     {
17        if ((choice != 'a') && (choice != 'b') &&
18        (choice != 'c'))
19        {  counter += 3;
20           flag = 0; }
21        else{
22           flag = 1;
23           counter *= 3;}     counter = counter * 3
24     }
```

!a = a==0

!0 = 0==0

```
25                                  12
26      for (int i = 0; i < counter;)
27      {
28          switch (*choice*)
29          {
30              case 'a' : ++i; cout << "one\n";
31                         break;
32              case 'b' : i+=2; cout << "two\n";
33                         break;                          i=i+3
34              case 'c' : i+=3; cout << "three\n";
35              default  : i+=4; cout << "four\n";         i=0+3
36                         break;                          i=3
37          }
38      }                                  i=i+4
39      return 0;                          i=4
40  }
```

Figure B4: C++ program code for Question 2

(a) From the **Figure B4**, there are 5 identified syntax errors in the code in the usage of the control structure codes (repetition and selection). Find the line of the code and write the correct code. Write your answer in **Table B3**. [5 marks]

Table B3: Answer for Question 2(a)

Line No.	Correction
15	while(!flag ~~&&~~ (counter % 2 == 0))
20	flag = 0; ~~}~~
21	else {
6	~~for (int i=0; i< counter; i++)~~ char choice;
28	switch (choice)

(b) Assuming that you have identified and found all the errors in Question 2(a), trace the program by filling in the values of the variables in the **Table B4** when the initial value is as shown below. [10 marks]

Table B4: Answer for Question 2(b)

flag	choice	counter
0	c	4
1	c	~~12~~

i	On display
0	-
3	three
7	four
10	three
14	four

flag	choice	counter
0	d	12
~~1~~ 0	~~d~~	~~36~~ 15

i	On display
0	-
4	four
8	four
~~12~~	four
16	four

3

QUESTION 3 **[15 MARKS]**

(a) **Figure B5** below shows the incomplete C++ code for calculating area for a house, complete the C++ code for the program and write your answer in **Table B5**. [5 marks]

```
#include <iostream>
using namespace std;

int main() {
    double length, width;

    //declares a character array named typeOfHouse capable of
    holding 60 characters.
char ____________(i);

    int numOfTimes,
    count = 0;

    cout << "Enter the type of house: ";
    ____________(ii)>> typeOfHouse;

    cout<<"How many times do you want to calculate the area? ";
    cin >> numOfTimes;

    while(_____________(iii))
    {
        cout << "Enter length and width of the rectangle: ";
        cin >> length >> width;
```

```
        double area = length * width;

        cout << "Area of the rectangle (" << length << " x "
        << width << ") for house type '" << typeOfHouse << "'
        is: ";

        //convert data type for the area to int
        cout << ______________(iv) << endl;

        //Updating of counter value (increase by 1) during each
        iteration
        ___________(v);
    }

    return 0;
}
```

Figure B5: C++ program code for Question 3(a)

Table B5: Answer for Question 3(a)

Question	Answer
(i)	char typeOfHouse [50]
(ii)	cin
(iii)	count >= 0
(iv)	static_cast <int> (area)
(v)	count ++

(b) **Figure B6** show a C++ program code that prompts the user to input two ~~numbers that are~~ **`multiplicationNum`** for the multiplication table and **`divisionNum`** for the division table. Additionally, the user should input a range limit (**`limit`**). The program displays the output of the multiplication table for **`multiplicationNum`** and the division table for **`divisionNum`** from 1 up to the specified **`limit`**. **Figure B7** shows an example of the expected output for the specified value of variables below. [10 marks]

From **Figure B6** and **B7**, answer the following questions:

(i). Write code segment for multiplication table. (5 marks)

(ii). Write code segment for division table. Use **static cast** to convert the output to **double** number. (5 marks)

```
#include <iostream>
using namespace std;

int main() {

    int multiplicationNum, divisionNum, limit;

    cout << "Enter a number for multiplication table: ";
    cin >> multiplicationNum;

    cout << "Enter a number for division table: ";
    cin >> divisionNum;

    cout << "Enter the limit: ";
    cin >> limit;

    cout << "Multiplication table for " << multiplicationNum <<
    " up to " << limit << ":" << endl;

    // (i) Write code segment for multiplication table.


    cout << "Division table for " << divisionNum << " up to "
    << limit << ":" << endl;

    // (ii) Write code segment for division table.

    return 0;
}
```

Figure B6: C++ program code for Question 3(b)

```
Enter a number for multiplication table: 10 [Enter]
Enter a number for division table: 12 [Enter]
Enter the limit: 5 [Enter]

Multiplication table for 10 up to 5:
10 x 1 = 10
10 x 2 = 20
10 x 3 = 30
10 x 4 = 40
10 x 5 = 50
```

```
Division table for 12 up to 5:
12 / 1 = 12
12 / 2 = 6
12 / 3 = 4
12 / 4 = 3
12 / 5 = 2.4
```

Figure B7: Example of the C++ program expected output for Question 3(b)

Answer space for Question 3(b)(i) and 3(b)(ii)

3.(b)(i)

```
int product;
for (int i=1 ; i <= limit ; i++)
{
    product = i * multiplicationNum;
    cout << multiplicationNum << " x " << i << " = " << product << endl;
}
```

3.(b)(ii)

```
double div_num;

for (int i=1 ; i <= limit ; i++)
{
    div_num = static_cast <double> (divisionNum) / i;
    cout << divisionNum << " / " << i << " = " << div_num << endl;
}
```

10

QUESTION 4 **[15 MARKS]**

(a) From the C++ program code in **Figure B8**, trace the output and write your answer in the answer column. [5 marks]

Line	Code	Answer
1	`#include <iostream>`	
2	`using namespace std;`	
3		
4	`int main() {`	
5	`int a = 0, b = 0, c = 0, d = 0;`	
6		
7	`cout << "\t\t" << "a" << "\t" << "b" << "\t" << "c" << "\t" << "d\n";`	0 0 0 0
8		
9	`cout << "initialize" << "\t";`	initialize
10		
11	`cout << a << "\t" << b << "\t" << c << "\t" << d << "\n";`	0 0 0 0
12		
13	`(a+=2)*(--a);`	a=2 1
14	`(--((++b)%=3));`	b=0
15	`((++c)/5.0);`	c=0.2
16	`++(d+=2);`	d=3
17		
18	`cout << "update" << "\t\t";`	update
19		
20	`cout << a << "\t" << b << "\t" << c << "\t" << d << "\n";`	2 0 0.2 3
21		
22	`return 0;`	
23	`}`	

Figure B8: C++ program code for Question 4(a)

(b) From the C++ program code in **Figure B9**, answer the question (i), (ii) and (iii) by writing the code segments for each task to produce the output in **Figure B10**. [10 marks]

(i). Write the code segment to print even numbers from 1 to 10 (3 marks)

(ii). Write the code segment to print odd numbers from 1 to 10 (3 marks)

(iii). Write the code segment to print prime numbers from 1 to 10 (4 marks)

Note: You are compulsory to apply a **LOOP** using **for/while/do...while AND** apply a **SELECTION** using **if/if...else/switch...case**.

```
#include <iostream>
using namespace std;
int main(){
     cout << "List numbers from 1 to 10\n";

     // Print numbers from 1 to 10
     for (int i = 1; i <= 10; ++i) {

          cout << i << " ";

          cout << "\n\nEven numbers from 1 to 10\n";
          // (i) Write the code segment to print even numbers
          from 1 to 10

          cout << "\n\nOdd numbers from 1 to 10\n";
          // (ii) Write the code segment to print odd numbers
          from 1 to 10

          cout << "\n\nPrime numbers from 1 to 10\n";
          // (iii) Write the code segment to print prime numbers
          from 1 to 10
     }
     return 0;
}
```

Figure B9: C++ program code for Question 4(b)

```
List numbers from 1 to 10
1 2 3 4 5 6 7 8 9 10

Even numbers from 1 to 10
2 4 6 8 10

Odd numbers from 1 to 10
1 3 5 7 9

Prime numbers from 1 to 10
2 3 5 7
```

Figure B10: C++ program output for Question 4(b)

Answer space for Question 4(b)(i), 4(b) (ii) and 4(b)(ii)

```
for (int i=1; i<=10; ++i)
{
(i) if (i % 2 ==0)
        cout << i << "  ";
}
```

~~for~~

3

```
for (int i=1; i<=10; ++i) {
(ii) if (i % 2 == 1)
        cout << i << "  ";
}
```

3

```
(iii) for( i=2 ; i<=i/2 ; i++)
{
    if(i %2 ==0)
        cout << i << "  ";
}
```

1+1.5

(SECJ1013) PROGRAMMING TECHNIQUE 1

SEM 1, SESSION 2023/2024

LAB EXERCISE 3 (4%)

INSTRUCTIONS TO THE STUDENTS

- This exercise must be done **individually**.
- Any form of plagiarism is **NOT ALLOWED**. Students who copied other students' assignments will get **ZERO** marks (both parties, students who copied, and students who shared their work).

SUBMISSION PROCEDURE

- Please submit this exercise no later than **December 21, 2023, Thursday (1 PM MYT)**.
- Only hardcopy is accepted for this submission with handwriting (at my office – n28a, level 2, room 02-31-01).

Fill your information here:

Name	Lau Yee Wen
Matric Number	A23CS0099
Year / Course	1 / SECPH
Section	03

Set 1

Line number	Correction Statement with the reason(s)
5	void display_Question (int); // function prototype need to declare the data type for variable in bracket and need a semicolon
6.	void yes_No (char k); // function prototype need to declare the data type for variable in bracket and need a semicolon. Also, the parameter must be a reference variable to pass the value of ans to main function.
7	int get_Status (char, char, char) // function prototype need to declare the data type for variable in the bracket and need a semicolon.
18.	for(int i=1; i<=3; i++){ // continue loop if (i==3), to display the question "Body temperature >= 38 degrees Celcius?\n" The loop needs to run 3 times.
19.	display_Question (i); // display_question (i) is not same with display_Question (i)
20.	if (i==1) // We need to check whether i is equal to 1 or not, instead of using single equal symbol to asign 1 to the variable i.
23.	else if (i==2) // We need to check whether i is equal to 2 or not, instead of using single symbol to asign 2 to the variable i.
35	switch (status) { // It is not a loop because while loop will lead to infinite loop
36	case 0: cout << "GREEN"; break; // change to switch condition
37	case 1: cout << "YELLOW"; break;
38	case 2: cout << "ORANGE"; break; // break out of the switch
39	default: cout << "RED"; break;
46	void display_Question (int q){ // the parameter q is a integer
47	switch (q){ // need a open bracket for switch statements
54	default: // We usually use default statements for last line of the switch statement
55	Celcius?\n"; break; // break out of the switch
56	} // void function cannot have return, not a return function
59	void yes_No (char &ans) { // need the ampersand (&) symbol to indicate that the parameter is a reference variable so that can pass the value by reference.

Line number	Correction statement with the reason(s)
62	cin >> ans ; // input use >> symbol
63	} while (ans != 'n' && ans != 'y') ; // the loop continues until a valid answer which is y or n is entered by the user
64	cout << "\n" ; // String is use double quote symbol

69
70
71
	if (rz == 'y') stt ;
	if (cc == 'y') stt ;
	if (f == 'y') stt ; // single equal symbol is used to asign a value to a variable, but here, we need to check and compare the values. For example, we are checking if rz is equal to y or not instead of asign y to rz.

SET 2

```
# include <iostream>
# include <string>
using namespace std;

// constants with the associated values
# define WASH 10.0   // the price of WASH service charge
# define VACUUM 7.0  // the price of VACUUM service charge
# define POLISH 15.0 // the price of POLISH service charge

// Task 6: List all function prototypes.
void setType (string&);
void setPackage (int&);
float wash (string);
float vacuum (string);
float polish (string);

// Task 7: (i) Call the functions from Task 1 until Task 5.
int main() {
    string carType; // car type variable
    int wsPkg; // car wash service package
    float totalCharge=0; // total service charge based on different wash service package
                         with different

    // call setType function with the parameter carType
    setType (carType);

    // call setPackage function with the parameter wsPkg
    setPackage (wsPkg)

// Task 7: (ii) totalCharge is the variable to hold the total service charge based on
  different wash service package with different car type (sedan or mpv).

switch (wsPkg) {
    case 1: totalCharge += wash(carType); break;
    case 2: totalCharge += vacuum(carType); break;
    default: totalCharge += polish (carType); break;
}
```

```
        cout << endl;
        cout << "Total service charge is " << totalCharge;
        return 0;
}

// Task 1: Function to set car type

void setType (string &type) {
        do
        {
            cout << " Enter car type (sedan /mpv) : " ;
            cin >> type ;
        } while (type != "sedan" && type != "mpv" ) ;
        cout << endl;
}

// Task 2: Function to set wash service package

void setPackage (int &pkg) {
        cout << "1. Basic " << endl;
        cout << " 2. Deluxe" << endl ;
        cout << " 3. Premium" << endl;

        do
        {
            cout << "Choose wash service package (1/2/3): " ;
            cin >> pkg ;
        } while (pkg != 3 && pkg != 2 && pkg != 1);
        cout << endl;
}

// Task 3: Function to determine exterior wash service charge based on car type

float wash (string type ) {
```

```
    float charge;
    if(type == "mpv")
        charge = 1.2* WASH;
    if (type == "sedan")
        charge = WASH;

    cout << " Wash service charge is " << charge << endl;
    return charge;
}

// Task 4 : Function to determine interior vacuum service charge based on car type

float vacuum (string type) {
    float charge;
    if(type == "mpv")
        charge = 1.05* VACUUM;
    if(type == "sedan")
        charge = VACUUM;

    cout << " Vacuum service charge is " << charge << endl;
    return charge;
}
// Task 5: Function to determine exterior polish service charge based on car type

float polish (string type) {
    float charge;
    if(type == "mpv")
        charge = 1.2* POLISH;
    if (type == "sedan")
        charge = POLISH;

    cout << " Polish service charge is " << charge << endl;
    return charge;
}
```