Aga Khan University Examination Board

Notes from E-Marking Centre on HSSC-II Computer Science Examination May 2017

Introduction

This document has been produced for the teachers and candidates of Higher Secondary School Certificate (HSSC-II) Computer Science. It contains comments on candidates' responses to the 2017 HSSC-II Examination, indicating the quality of the responses and highlighting their relative strengths and weaknesses.

E-Marking Notes

This includes overall comments on candidates' performance on every question and *some* specific examples of candidates' responses which support the mentioned comments. Please note that the descriptive comments represent an overall perception of the better and weaker responses as gathered from the e-marking session. However, the candidates' responses shared in this document represent some specific example(s) of the mentioned comments.

Teachers and candidates should be aware that examiners may ask questions that address the Student Learning Outcomes (SLOs) in a manner that requires candidates to respond by integrating knowledge, understanding and application skills they have developed during the course of study. Candidates are advised to read and comprehend each question carefully before writing the response to fulfill the demand of the question.

Candidates need to be aware that the marks allocated to the questions are related to the answer space provided on the examination paper as a guide to the length of the required response. A longer response will not in itself lead to higher marks. Candidates need to be familiar with the command words in the SLOs which contain terms commonly used in examination questions. However, candidates should also be aware that not all questions will start with or contain one of the command words. Words such as 'how', 'why' or 'what' may also be used.

General Comment

In general, questions related to database objects, data types, primary key, sorting, query in design view, conversion of arithmetic expression into C program, building logic using selection statement and remainder operator were well attempted. However, questions based on filtering, advantages of C programming, functions in C programming, file management in C programming, validation in C programming, calculating maximum value and visual basic programming questions were generally not well attempted.

Detailed Comments:

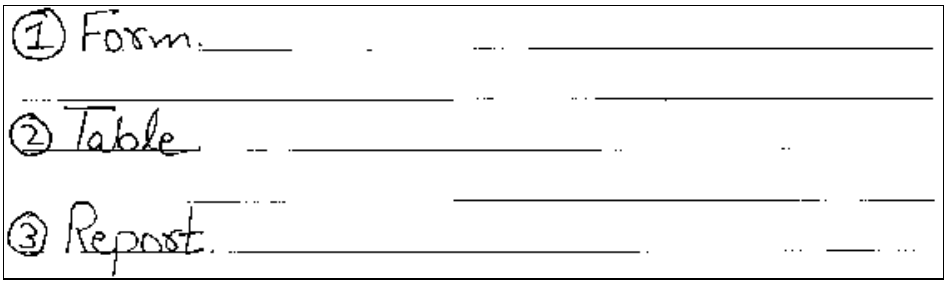
Constructed Response Questions (CRQs)

Question 1:

A database object is used to store or reference data.
Name any THREE database objects.

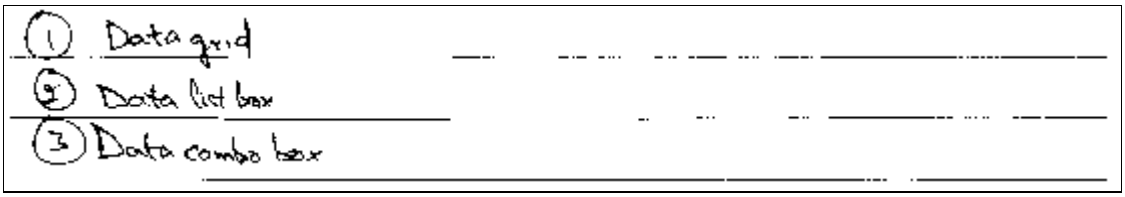
Better responses wrote the correct names of database objects such as tables, queries, forms, reports and relationship.

Example:



Weaker responses showed that candidates had no idea about database objects and wrote random names.

Example:



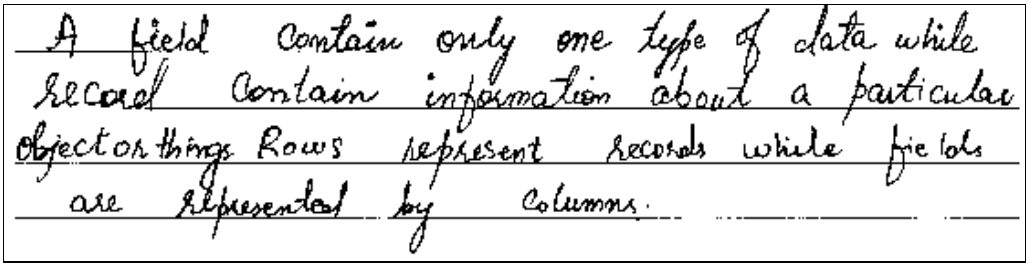
Question 1b:

A database table consists of fields and records.

Describe the major difference between a field and a record.

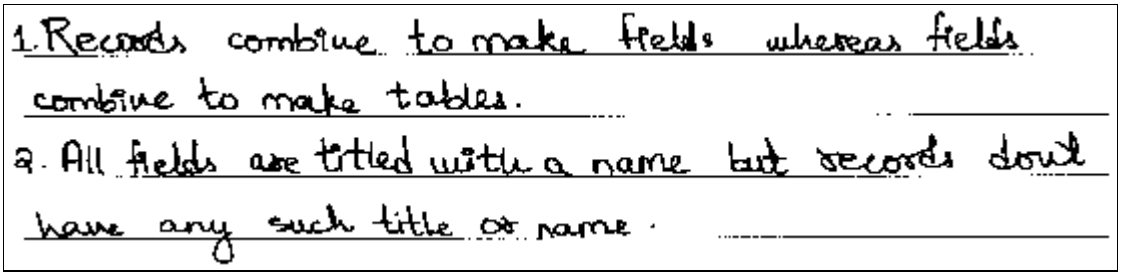
Better responses differentiated between a field and record on the basis of their appearance in database table, e.g. field is column in a table while record is row in a table. Also, such responses differentiated between fields and records on the basis of data they hold, for instance, a record contains all the data about an entity while a field contains a single piece of data for all attributes.

Example:



Weaker responses were not able to write the technical difference between field and record. These responses wrote irrelevant/ incorrect differences such as a field has heading but a record has no heading/ field consists of records whereas records consists of fields/ data type is declared on the basis of field but in record, data type is already declared.

Example:



Question 2:

Omar has created a database table that is shown below.

Student ID	First Name	Last Name	Class	Date of Birth	Fees Paid
S101	Ali	Khan	11	10/09/1999	Yes
S102	Jahangir	Ahmed	11	03/15/1998	No
S103	Sumaira	Usmani	12	06/26/1997	No
S104	Nadia	Zafar	12	03/15/1998	Yes
S105	Ali	Ahmed	12	11/20/1997	Yes

Write the CORRECT data types he selected for the fields given below:

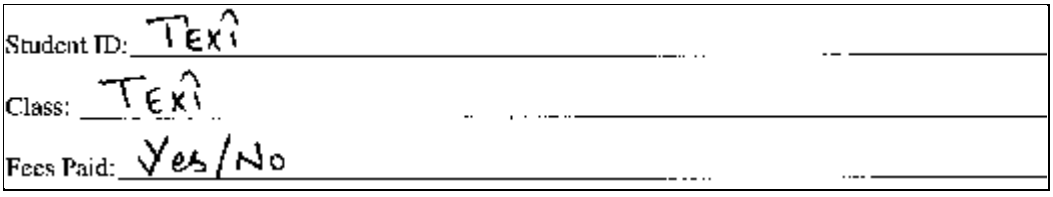
Student ID:
Class:
Fees Paid:

Write the name of the field from the given table which should be selected as a primary key. Give a valid reason to support your answer.
Primary Key:
Reason:

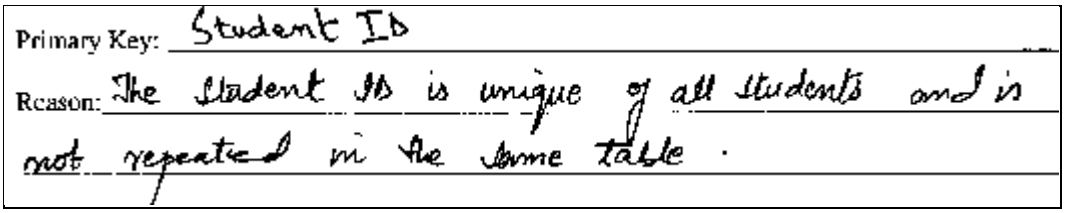
Better responses depicted that candidates had practiced the creation of tables and selection of appropriate data types in MS Access. Moreover, such responses showed that candidates had the ability to choose correct primary key for a table in MS Access.

Example:

(a)



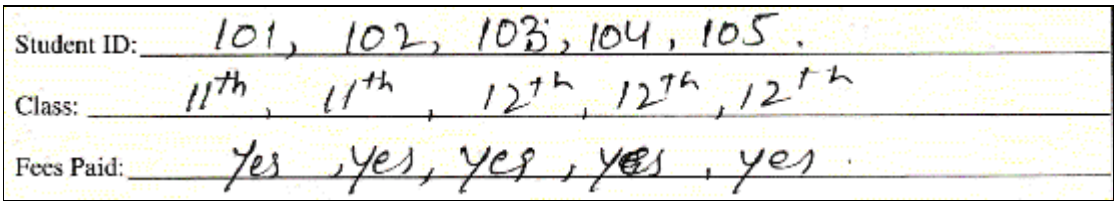
(b)



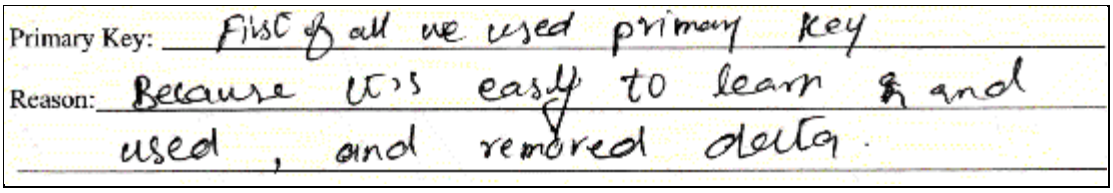
Weaker responses mostly selected the data types based on the names of fields and ignored the data inside the table, e.g. most of these responses wrote numeric data type for **Student ID** and currency data type for **Fees Paid**. Some of these responses wrote the data itself for each data field rather than writing the data types. However, most of these responses were able to identify the primary key for the given table but only few of them wrote the reason to select primary key.

Example:

(a)



(b)

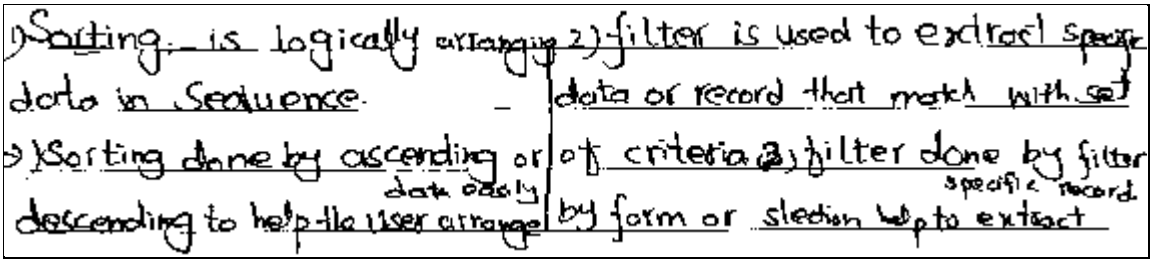


Question 3a:

Differentiate between the two database options filter and sorting.

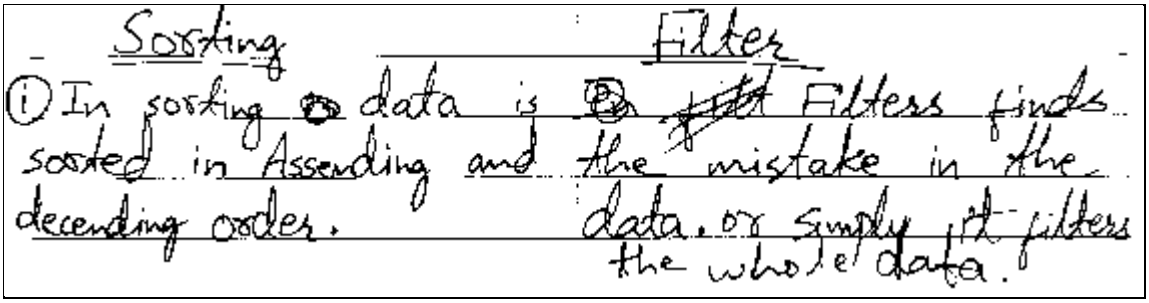
Better responses differentiated between filter and sorting on the basis of their functions, e.g. sorting is the process of organising a set of records in a particular order while filtering is the process of displaying records that meet a given condition.

Example:



Weaker responses mostly showed clarity in concepts regarding sorting. However, such responses depicted that candidates had unclear concept about filtering and were not able to write about it. Hence, they were not able to differentiate.

Example:



Question 3b:

Consider the given table.

Microsoft Access - [Employees : Table]

File Edit View Insert Format Records Tools Window Help

Employee ID	Employee Name	Salary	City	Birthdate
1	Kamal Ahmed	$25,000.00	Karachi	12/2/1978
2	Asad Shah	$40,000.00	Islamabad	3/5/1976
3	Ali Khan	$30,000.00	Lahore	4/6/1980
4	Syed Faisal	$34,000.00	Karachi	1/1/1979
5	Amjad Khan	$50,000.00	Karachi	12/2/1978
6	Kamran Khan	$70,000.00	Lahore	1/27/1977
(AutoNumber)				

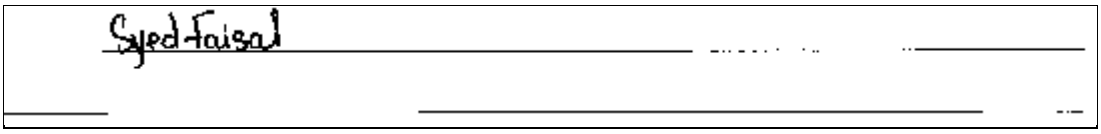
i. If the table is sorted in ascending order by "Employee Name", then which employee's record will be mentioned at the end?

ii. If the table is sorted in descending order by "Birthdate", then which employee's record will be mentioned in the beginning?

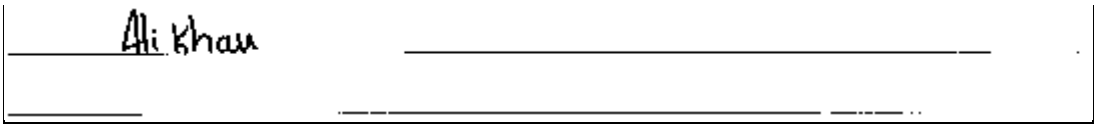
Better responses demonstrated good understanding of applying the concept of ascending and descending sort on the given data and, hence, were able to correctly identify 'Syed Faisal' for part **(i)** and 'Ali Khan' for part **(ii)**.

Example:

(i)



(ii)

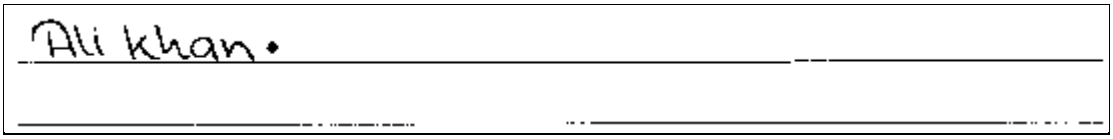


Weaker responses showed either candidates were confused in between the working of ascending and descending sort or they did not read and understand the question. Most of such responses wrote 'Ali Khan' in part **(i).**

Moreover, such responses applied the descending sort based on the year of birth, i.e. 1976 rather than sorting them on the basis of age and wrote 'Asad Shah' in part **(ii)**.

Example:

(i)



(ii)

Question 4:

A database named 'CarpetRecords' was setup to store data of carpets such as type, colour, material, quantity, price, etc. Some records from the database are shown below.

CarpetRecords						
Batch ID	**Type of Carpet**	**Quantity**	**Colour**	**Material Used**	**Made in**	**Price in Rupees**
C100	Woven	5	Black	Wool	Pakistan	18000
C101	Needle Felt	3	Brown	Polyester	China	12500
C102	Needle Felt	5	Blue	Wool	Pakistan	20000
C103	Tufted	7	Green	Nylon	Turkey	13000
C104	Woven	6	Blue	Wool	China	17000
C105	Needle Felt	2	Black	Wool	Turkey	22500

i. Consider the following query in design view of MS Access.

Field:	Batch ID	Type of Carpet	Colour	Made in	Price in Rupees
Table:					
Sort:					
Show:	☐	☑	☑	☐	☑
Criteria:			= "Black"		
or:				= "Pakistan"	

Show the output when the given query is executed.

Space for output

ii. Complete the query in design view shown below to select and show the Batch ID, Type of Carpet, Quantity, Made in and Price in Rupees of all types of carpets with 5 or more quantity.

Field:					
Table:					
Sort:					
Show:	☐	☐	☐	☐	☐
Criteria:					
or:					

Better responses exhibited good understanding of queries in design view and applied this concept on the given scenario to determine the output and constructed the query in design view to show the data as per the criteria given in the question.

Example:

(i)

Space for output

Woven	Black	18000
Needle Felt	Blue	20000
Needle Felt	Black	22500

(ii)

Field:	Batch ID	Type of Carpet	Quantity	Made in	Price in Rupees
Table:					
Sort:					
Show:	☑	☑	☑	☑	☑
Criteria:			>=5		
or:					

Weaker responses depicted that most of the candidates misunderstood part **(i)** of this question and wrote only one record in the output rather than writing all the records that were meeting the given criteria.

However, most of such responses completed the query in part **(ii)** and generally secured 1 or 2 marks out of 3 in this part. To complete the given query, these responses wrote the names of fields and ticked the boxes to show the fields but either they were not able to apply the criteria or were not able to write the field names according to the sequence given in the question.

Example:

(i)

Space for output

Type of carpet	Colour	Price in rupees
Woven	Black	18000

(ii)

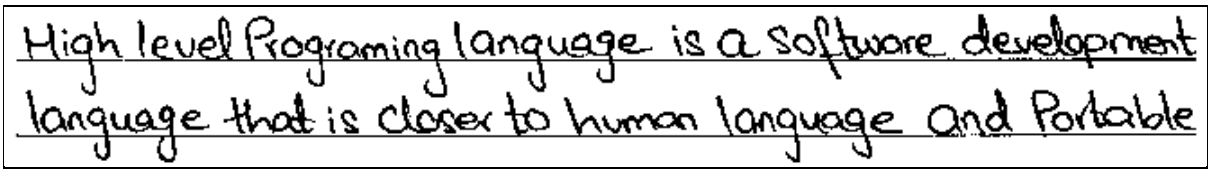
Field:	Batch ID	Type of carpet	Made	Price in Rs	Quantity
Table:					
Sort:					
Show:	✓	✓	✓	✓	✓
Criteria:					="SS=1"
or:					

Question 5a:

Define the term 'high level programing language'.

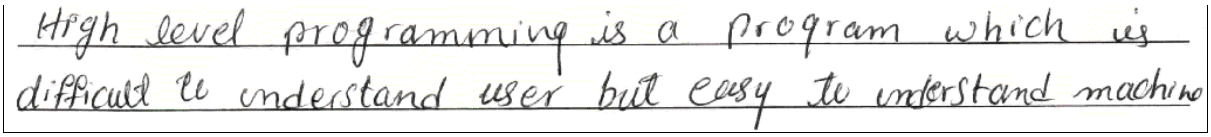
Better responses wrote the correct definition of high level programming, e.g. it is easy for humans to understand/ it is human-oriented programming language.

Example:



Weaker responses wrote the definition of low level programming language instead of high level programming language which showed that candidates were confused between these two programming languages.

Example:

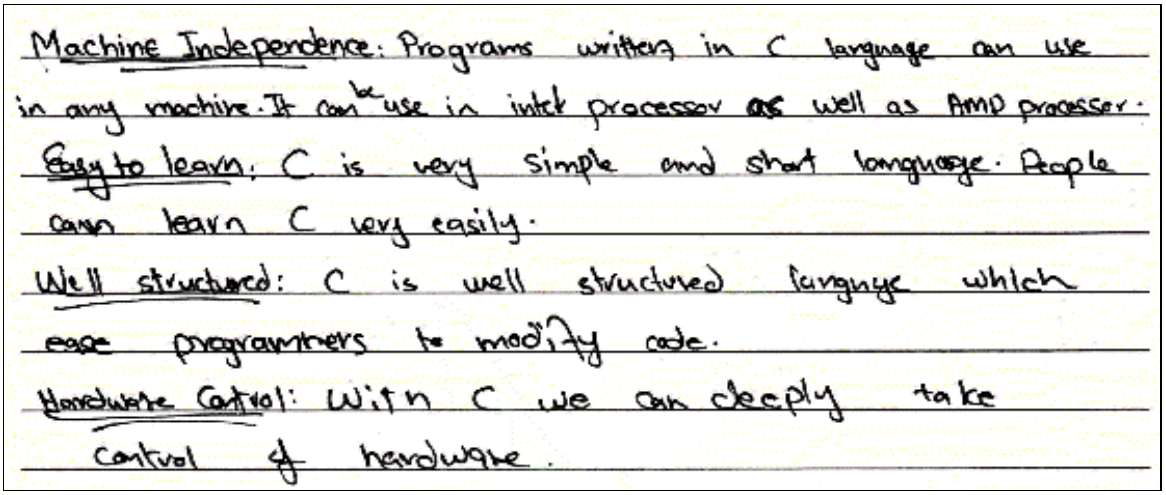


Question 5b:

Describe FOUR advantages of C programming language over other programming languages.

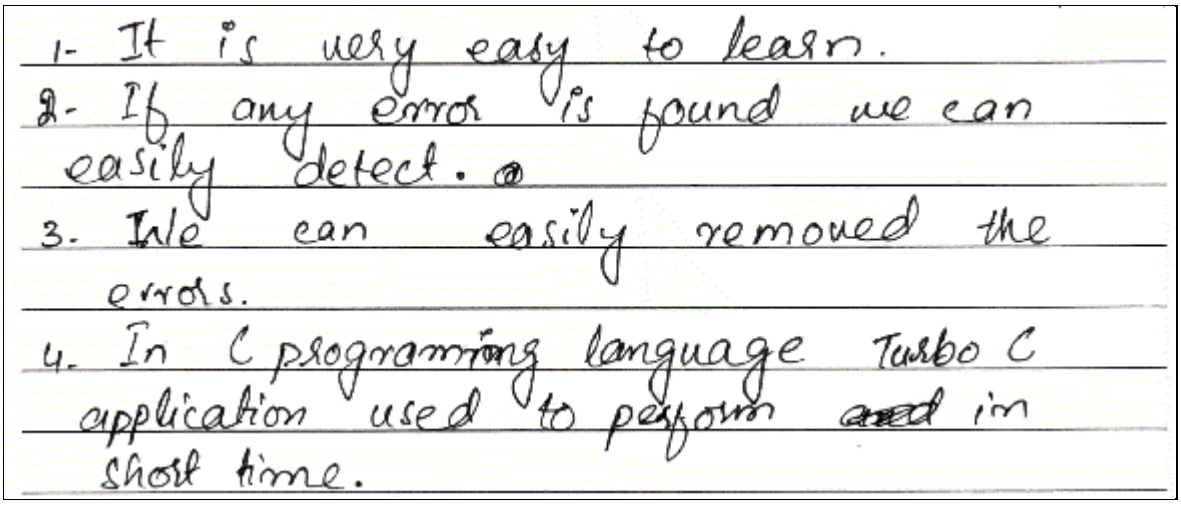
Better responses wrote those advantages of C programming language which are comparable with other programming languages. For instance, it is a well-structured language; it is supported by multiple platforms; it is a compiled programming language; it is a middle-level language, etc.

Example:


Machine Independence: Programs writtern in C language can use in any machine. It can be use in intel processor as well as AMD processer.
Easy to learn: C is very simple and short language. People can learn C very easily.
Well structured: C is well structured language which ease programmers to modify code.
Hardware Control: With C we can deeply take contrul of hardware.

Weaker responses showed that candidates understood the question but they were not able to write the technical advantages of C programming language. Instead of that, they wrote general advantages of it such as errors can easily be detected; errors can be removed easily; it provides built-in functions; it provides operators and data types; it offers different keywords, etc.

Example:


1- It is very easy to learn.
2- If any error is found we can easily detect.
3. We can easily removed the errors.
4. In C programming language Turbo C application used to perform and in short time.

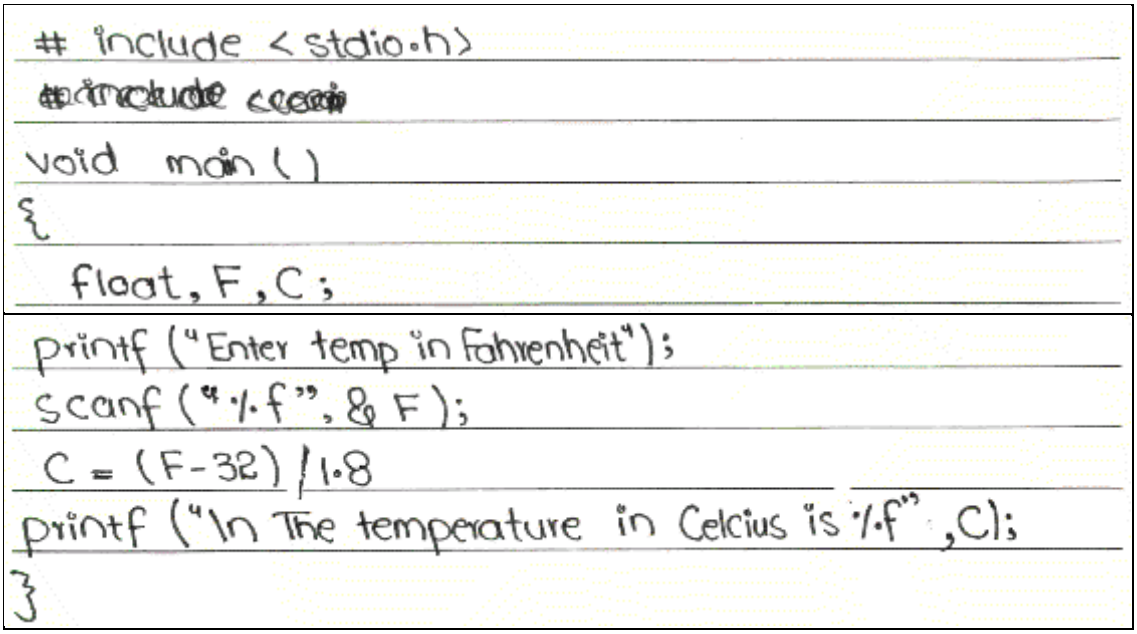
Question 6:

A student needs a computer program to convert temperature from degrees Fahrenheit (°F) to degrees Celsius (°C). Write a program using C language for the student that would meet the following requirements:

- It should take degrees Fahrenheit value as input.
- The degrees Fahrenheit value should be converted into degrees Celsius using the formula °C = (F – 32) ÷ 1.8
- It should display degrees Celsius value with appropriate message in output.

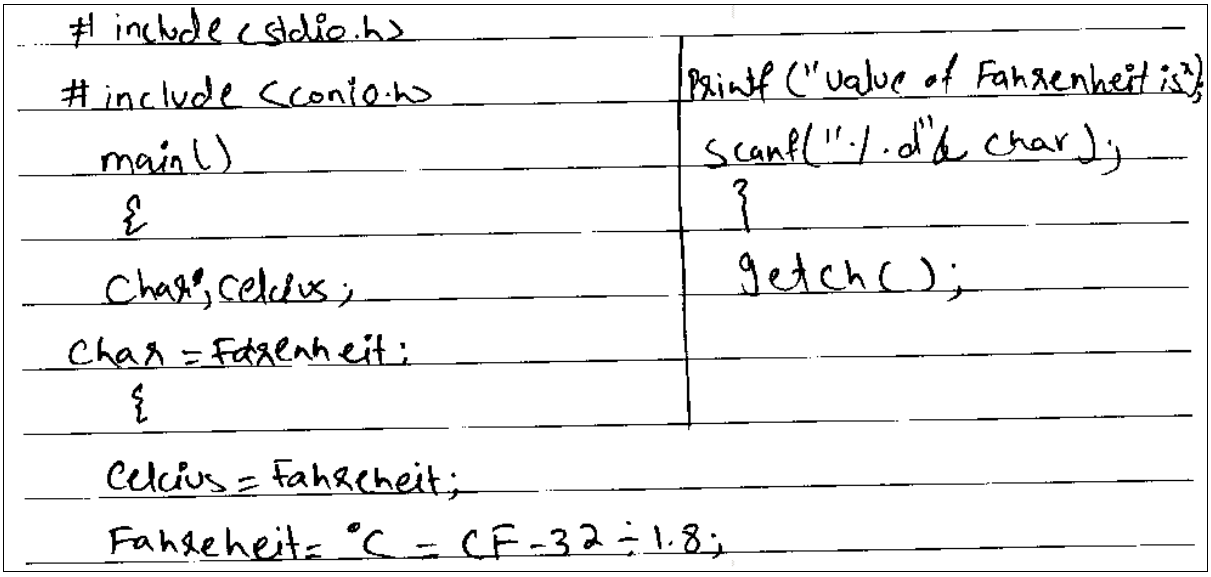
Better responses demonstrated good understanding of declaring variables, taking input values and converting a given mathematical expression into C language program.

Example:



Weaker responses depicted that candidates attempted this C programming question without practice due to which they were even not able to declare variables with appropriate data types. Similarly, such responses used mathematical operators while writing C program which is incorrect.

Example:

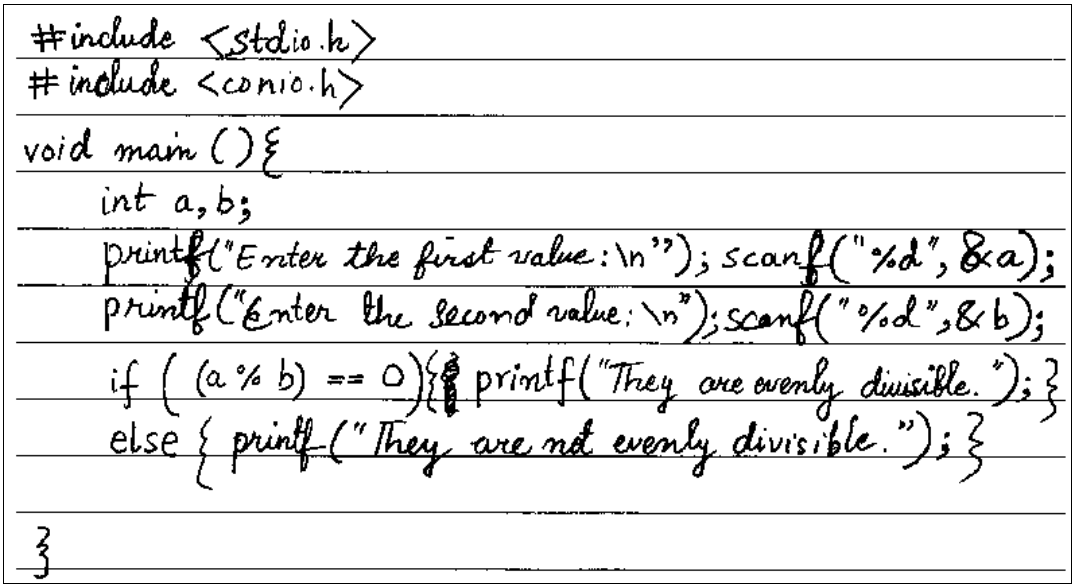


Question 7:

Write a program using C language which will take two integer values as input, test these two values to check if the first value is evenly divisible (no remainder left) by the second value and, finally, display appropriate output message.

Better responses wrote C program without syntax or logic errors. Such responses understood the requirement of question and built the logic in their program accordingly.

Example:

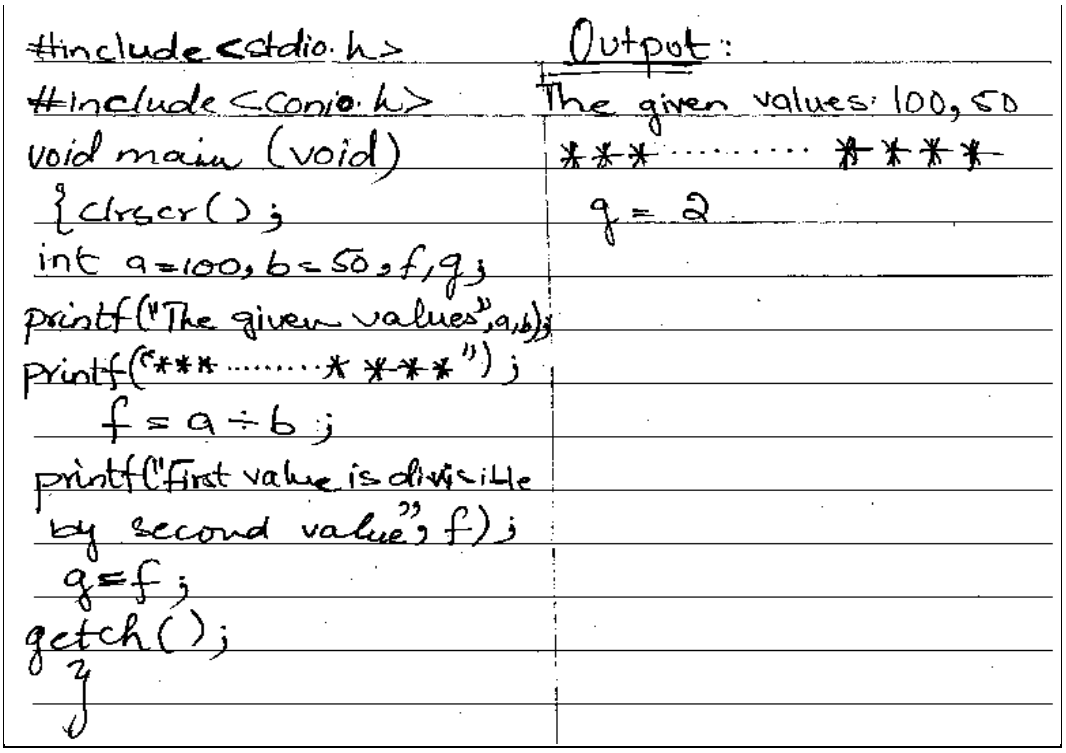


```
#include <stdio.h>
#include <conio.h>
void main (){
    int a,b;
    printf("Enter the first value:\n"); scanf("%d",&a);
    printf("Enter the second value:\n"); scanf("%d",&b);
    if ( (a % b) == 0){{ printf("They are evenly divisible."); }
    else { printf("They are not evenly divisible."); }

}
```

Weaker responses showed that candidates were unable to build logic in programming for given problem in which they were supposed to use remainder operator (%) to determine whether a number is even or odd. Most of such responses used division to solve this problem which is incorrect.

Example:



```
#include<stdio.h>
#include<conio.h>
void main (void)
 {clrscr();
int a=100, b=50, f, q;
printf("The given values",a,b);
printf("***.....****");
    f = a ÷ b ;
printf("First value is divisible
 by second value", f);
  q=f;
getch();
}
```

Output:

```
The given values: 100, 50
***.....****
 q = 2
```

Question 8a:

State the purpose of the following elements of functions in programming.

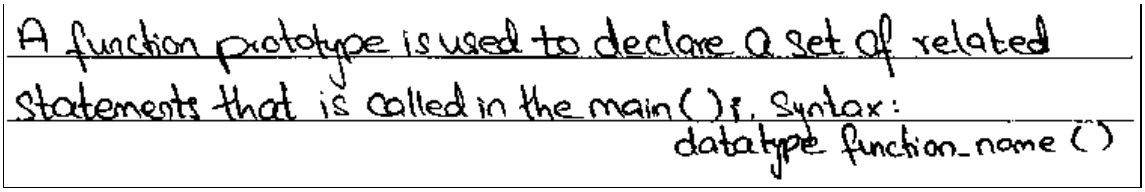
i. Function prototype
ii. Function call
iii. Function definition

Note: Last year, a programming based question was given in the paper from the topic of functions and most of the students performed average to below average in it. This year, question 8 is testing the theoretical concepts of students from the topic of functions in C programming and their overall response is almost the same. Teachers need to conduct more lectures, provide more worksheets and other reference material to students to improve their performance in future.

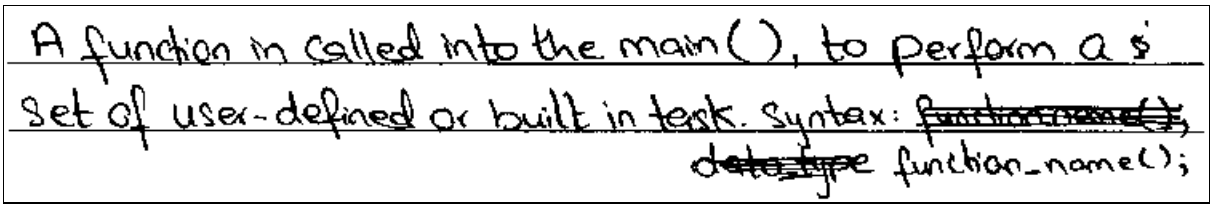
Better responses depicted good understanding of functions in programming and stated the correct purpose of function prototype, function call, and function definition.

Example:

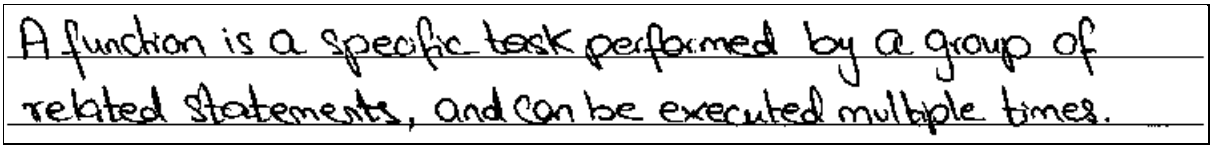
(i)



(ii)



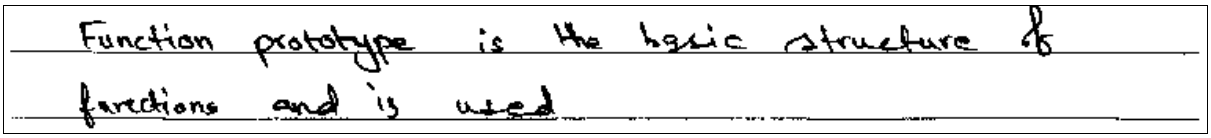
(iii)



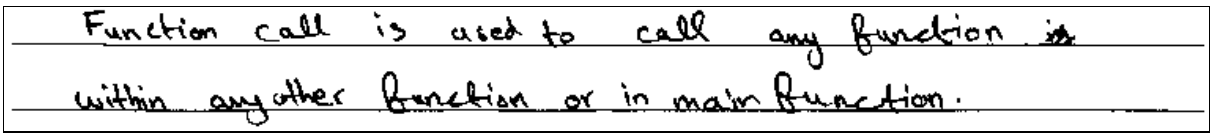
Weaker responses mostly stated the purpose of only function call but such responses stated incorrect purpose of function prototype and function definition.

Example:

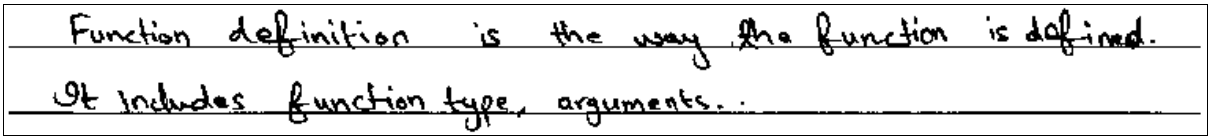
(i)



(ii)



(iii)

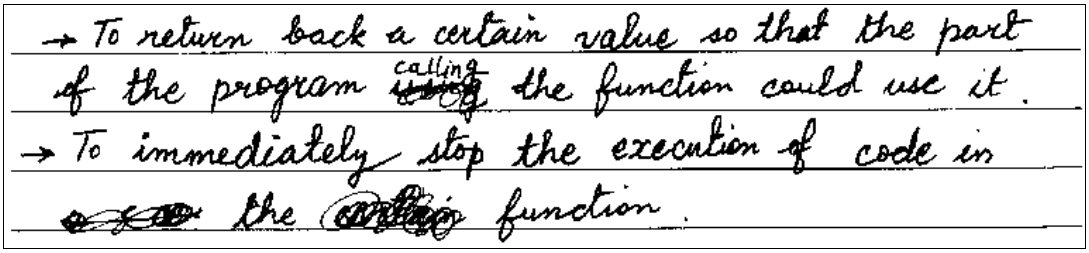


Question 8b:

Write the TWO purposes of return statement in C functions?

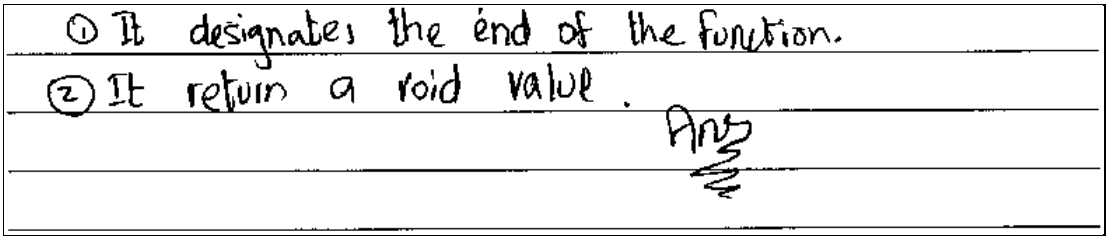
Better responses stated correct purposes of return statement in C function, e.g. it causes an immediate exit from the function; it causes program execution to return to the calling code; it is used to return a value.

Example:



Weaker responses were not able to write the purpose of return statement in technical terms, e.g. return statement tells the end of the function; it returns a void value; it executes a program; it tells the next statement that whether the previous statement was successful or failed, etc.

Example:



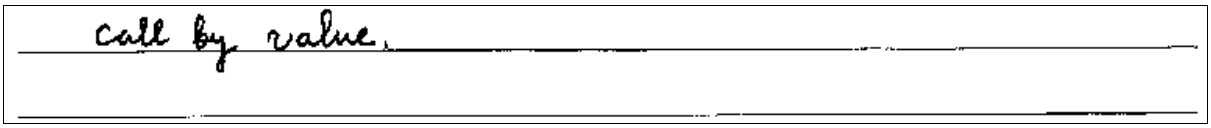
Question 8c:

In a computer language, there are two ways in which arguments can be passed to a function, call by value and call by reference.

Which of these is the default way in C language for passing arguments to a function?

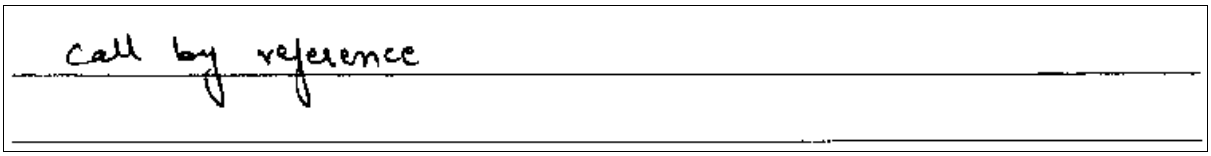
Better responses showed good knowledge about the default way to pass arguments to a function and wrote 'call by value' to answer this question.

Example:



Weaker responses were not able to answer this basic question about passing arguments to a function. Such responses did not know the purpose of using these two ways, i.e. 'call by value' and 'call by reference'.

Example:

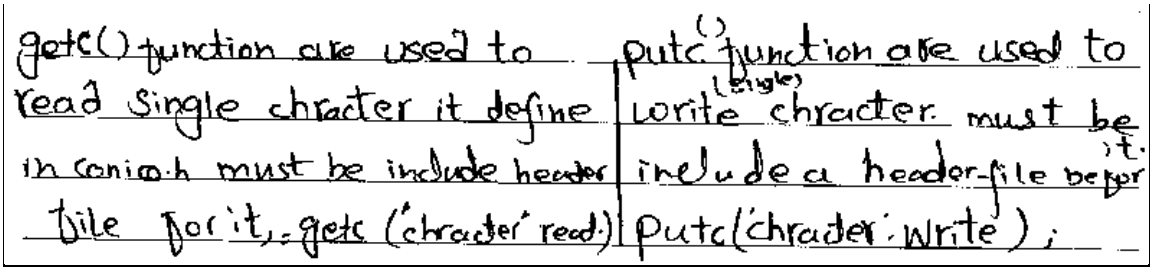


Question 9a:

Differentiate between getc() and putc() functions.

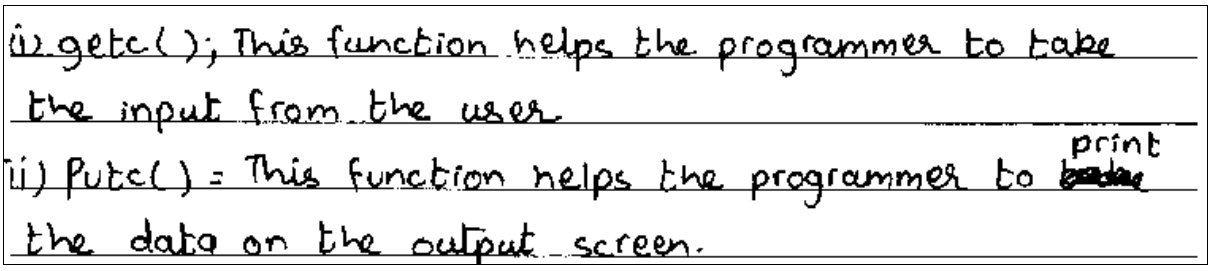
Better responses exhibited good understanding of the built-in file management functions in C language and differentiated the given functions technically, i.e. getc() function is used to read single character from a text file and putc() function is used to write a single character in a text file.

Example:



Weaker responses mostly did not know that these functions are built-in file management functions in C language. Due to this reason, such responses did not refer file management to differentiate between the given functions, instead of that, they were referring program and user.

Example:

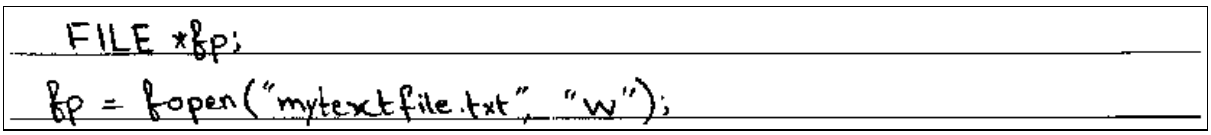


Question 9b:

Write the C language statement to open a text file named mytextfile.txt in write mode.

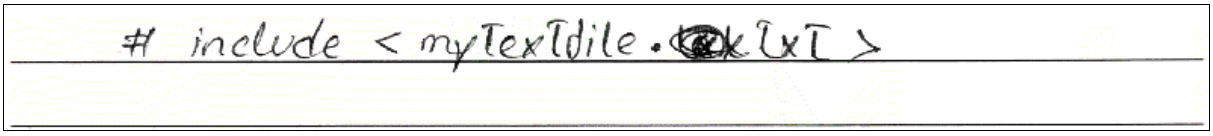
Better responses depicted that candidates had practiced file handling in C language and wrote the correct C language statement to open a text file.

Example:



Weaker responses showed lack of practice of file handling in C language due to which they were not able to write the correct statement to open file named mytextfile.txt.

Example:



Extended Response Questions (ERQs)

The following questions offered a choice between part **a** and **b**.

Question 10a:

Write a program in C language that would meet the following criteria.

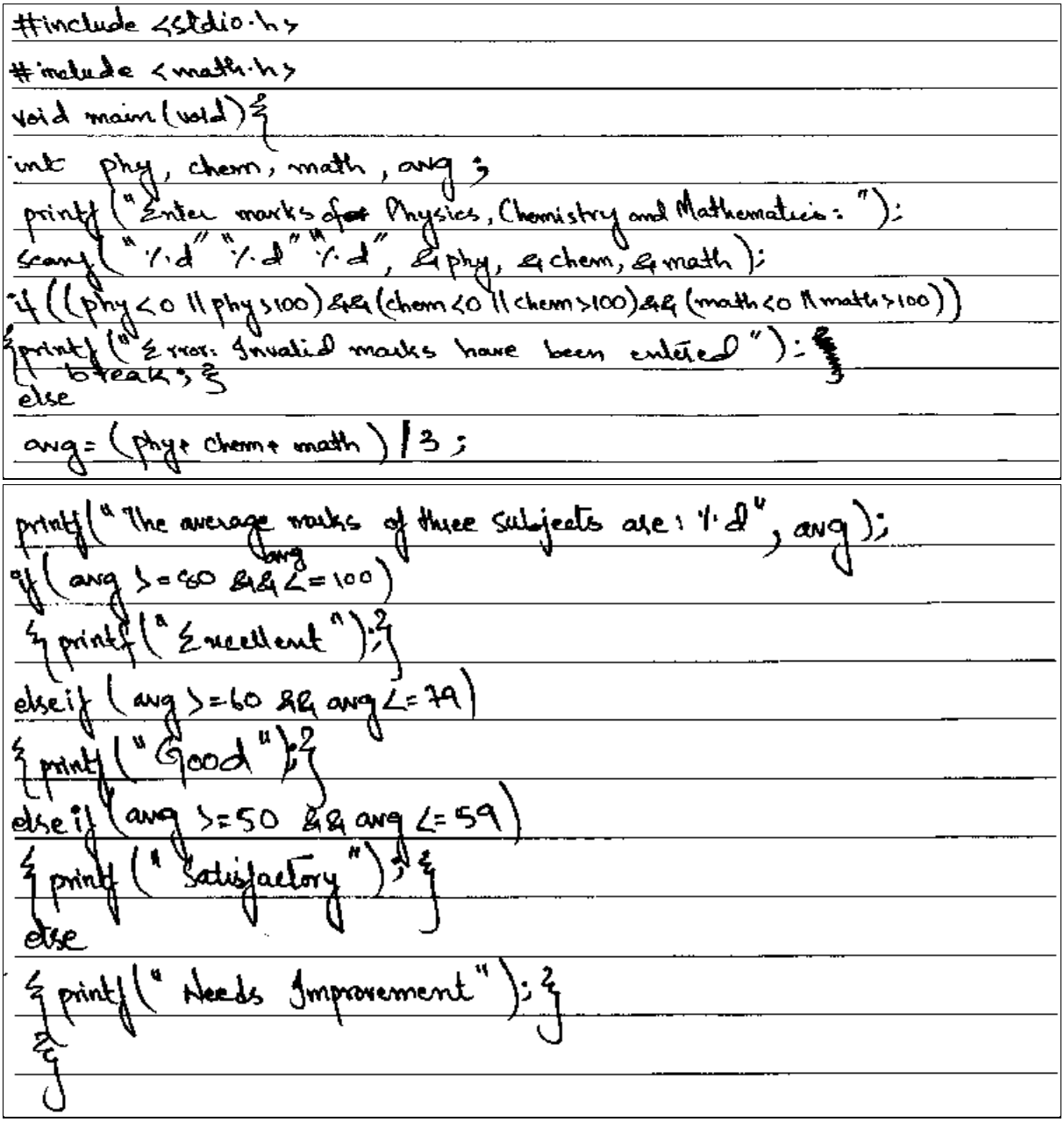
- It should input marks for three subjects, i.e. Physics, Chemistry and Mathematics.
- The marks should be between 0 and 100 (both inclusive) and if any marks value entered is outside of this range, then an error message should be displayed.
- It should calculate and display the average marks of the three subjects.
- It should display relevant remarks depending upon the average marks of the three subjects. (See table)

Average Marks	Remarks
80 to 100	Excellent
60 to 79	Good
50 to 59	Satisfactory
0 to 49	Needs Improvement

(**Note:** Most of the candidates opted part **a** of this question and they performed well.)

Better responses understood the given problem well and built the logic to solve this problem based on the given criteria such as marks should be between 0 and 100 (used IF statement to check this condition) and displaying remarks depending upon the given range (use IF statement to check the marks range and showing remarks).

Example:



```
#include <stdio.h>
#include <math.h>
void main(void){
int phy, chem, math, avg;
printf("Enter marks of Physics, Chemistry and Mathematics: ");
scanf("%d" "%d" "%d", &phy, &chem, &math);
if((phy<0 || phy>100) && (chem<0 || chem>100) && (math<0 || math>100))
{printf("Error: Invalid marks have been entered");
break;}
else
avg=(phy+chem+math)/3;
printf("The average marks of three subjects are: %d", avg);
if(avg>=80 && avg<=100)
{printf("Excellent");}
else if(avg>=60 && avg<=79)
{printf("Good");}
else if(avg>=50 && avg<=59)
{printf("Satisfactory");}
else
{printf("Needs Improvement");}
}
```

Weaker responses mostly understood the question but due to lack of practice and weaker concepts of programming control statements and loops, they were not able to build the logic required to solve the given problem. Such responses used loop statements instead of selection statement (IF…ELSE); calculated average in the end and did not use the average value to compare and show remarks; compared two constant values using IF statement (60 <= 70) rather than comparing the value of average variable with numeric values; did not write the programming statement for validation check (marks should be from 0 to 100).

Example:

```
#include <stdio.h>
#include <conio.h>
Void main (void)
{ clrscr( );
int physics, chemistry, mathematics;
printf ("The marks of subsects");
scanf("%d %d %d", &physics, &chemistry, &mathematics);
for (physics=2; physics ≥ 50; physics ++):
{printf (" satisfactory ");
```

```
for (chemistry = 4; chemistry ≥ 60; chemistry++)
{ printf (" Good"); }
for (mathematics = 4; mathematics ≥ 80; mathematics++)
{ Printf ("Excellent"); } }
  getch( );
  }
```

Output:

physics : satisfactory : 55
chemistry : Good : 71
Mathematics : Excellent : 90

Question 10b:

A gym trainer wants to calculate the average weight of the gym members.
Write a program using C language that would input weight of 35 gym members. The program should also calculate and display the maximum weight and average weight values of all gym members.
(**Note:** Use the For loop for repeating the code.)

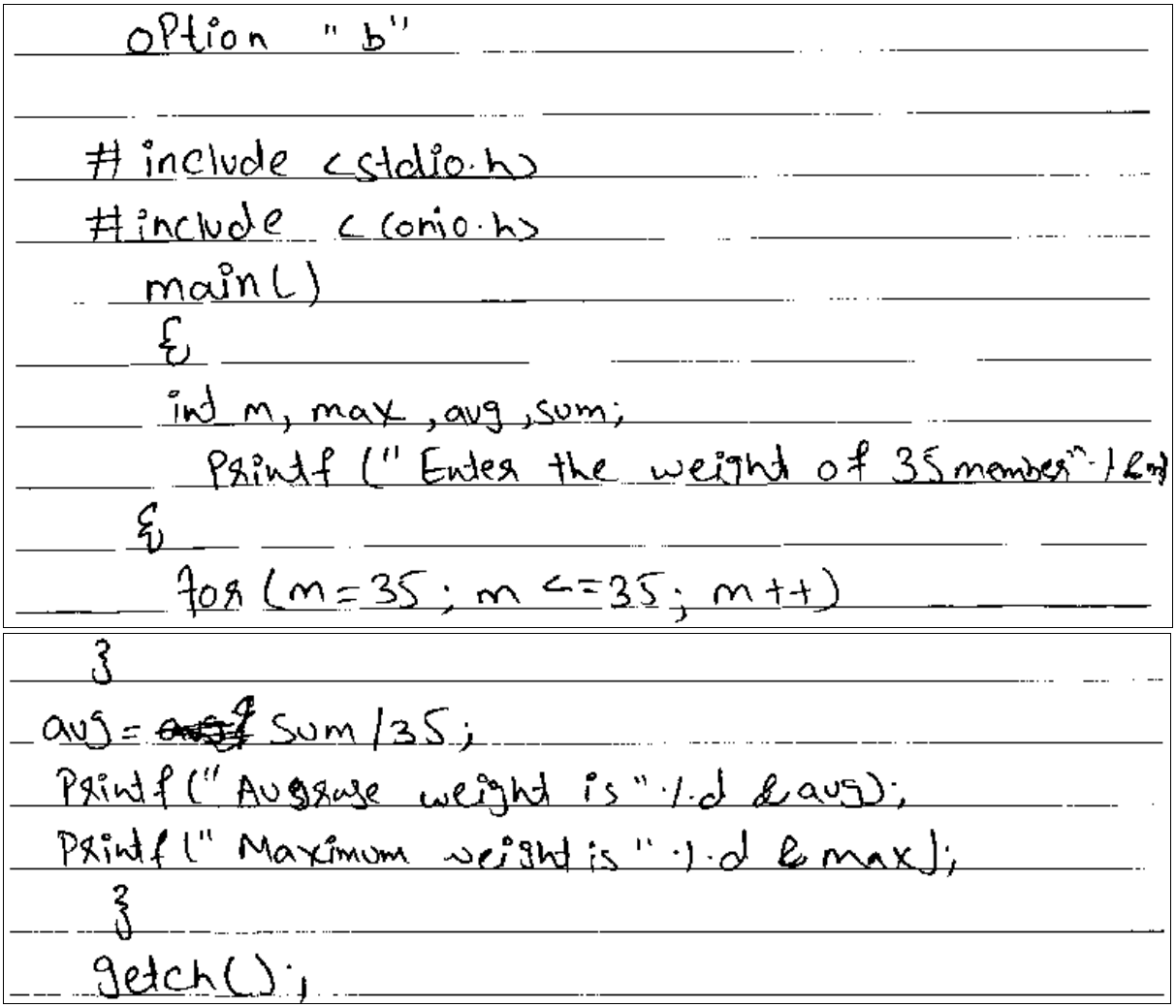
Better responses declared variables with appropriate data types to store weight of gym members and used for loop to repeat the code for 35 gym members. Furthermore, these responses performed totaling inside the loop and calculated average outside the loop. Moreover, such responses used IF…ELSE statement to calculate maximum weight value and displayed average and maximum weight values in output.

Example:

```
b). #include<stdio.h>
   #include<conio.h>
   void main()
   {
    float w,c, avg, max, sum;
     max= sum=0;
    printf("Enter weight:");
    for(c=1, c<=35, c++)
     {
       scanf("%f", &w);
if(w>max)
max=w;
sum=sum+w;
}
avg=sum/35;
printf("Maximum weight is %f", max);
printf("Average weight is %f", avg);
getch();
}
```

Weaker responses did not understand the question due to which they were not able to build the logic required to solve the given problem. Such responses were not able to initialise the loop with correct initial and final values; did not use IF…ELSE statement to calculate maximum weight value; did not perform totaling and average of weight values.

Example:



```
option "b"

#include <stdio.h>
#include <conio.h>
main()
{
int m, max, avg, sum;
Printf ("Enter the weight of 35 members"
{
for (m=35; m<=35; m++)
}
avg = ~~avg~~ sum/35;
Printf ("Average weight is "%d &avg);
Printf ("Maximum weight is "%d &max);
}
getch();
```

Note: Less than 10% candidates solved questions 11-16 that were based on VB.NET programming. It was good to see that candidates tried to attempt Visual Basic Programming Section, however, percentage of average marks of students in this section is below than 10 which is due to lack of practice. To improve the score in this section, candidates and teachers are advised to go through the SLOs of Visual Basic Section and then study with the help of reference books and links mentioned in the syllabus and solve the past paper questions based on this section.