Aga Khan University Examination Board

Notes from E-Marking Centre on HSSC-II Computer Science Examination April/ May 2019

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 2019 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 Comments

In general, questions related to database objects (forms, reports and queries), types of relationships in database, primary key, query design in MS Access, high level and low level programming langauges, escape sequences in C programming, for loops in programming, user-defined functions and maximum and minimum values were well attempted. However, questions based on find and replace function in MS Access, sorting of alphabets in MS Access, C program compilation process, file handling in C language, using loop within a loop, counting the occurrence of something in C language and Visual Basic (VB) programming questions were generally not well attempted.

Detailed Comments:

Constructed Response Questions (CRQs)

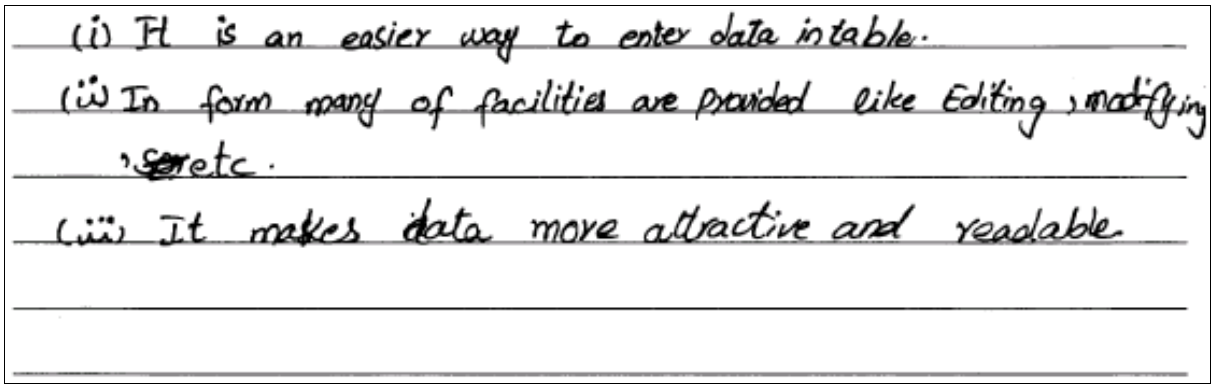
Question 1a:

Forms are preferred to enter data into a database table instead of entering data directly into the table.

Write any THREE reasons to support the given statement.

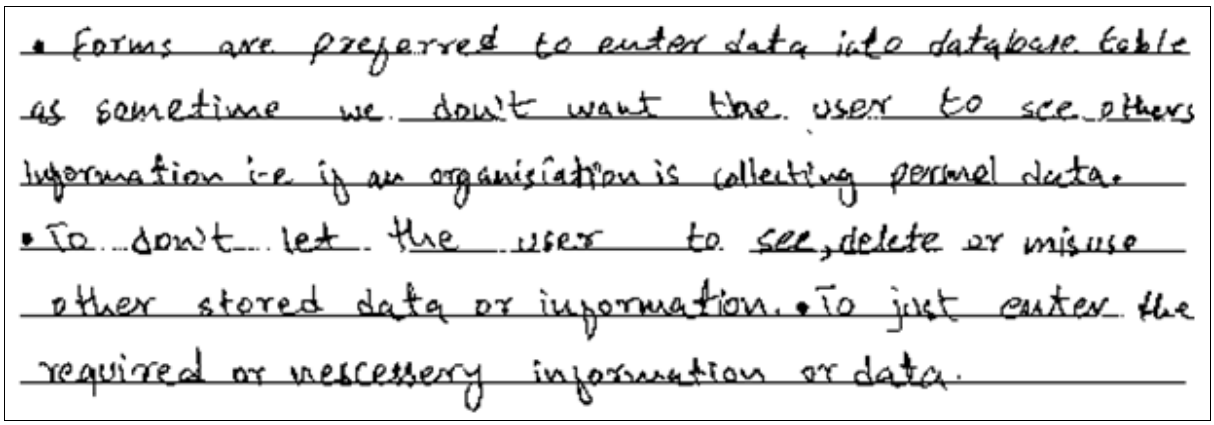
Better responses depicted that candidates had a good understanding of forms and wrote relevant reasons to support the statement given in the question. These responses included: records can be read easily in forms; it took less time to enter data in forms; forms are easier way to enter data in tables; easy to modify data via forms.

Example:


(i) It is an easier way to enter data in table.
(ii) In form many of facilities are provided like Editing, modifying, Sereetc.
(iii) It makes data more attractive and readable.

Weaker responses showed that candidates had a poor understanding of forms and wrote irrelevant or general reasons to justify the statement given in the question. These responses included: forms are used so that user cannot see other information; user cannot delete data; to enter required data; it has variety of options; it is easy to use; data entered in forms is automatically moved to tables.

Example:


• Forms are preferred to enter data into database table as sometime we dowt want the user to see others information i.e if an organisation is collecting personel data.
• To dowt let the user to see, delete or misuse other stored data or information. • To just enter the required or nescessery information or data.

Question 1b:

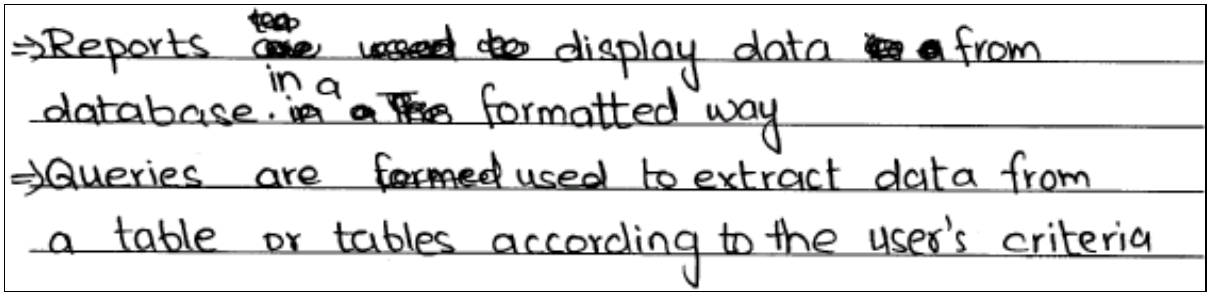
Define the following database terms.

- Reports
- Queries

(**Note**: Most of the candidates attempted this question well.)

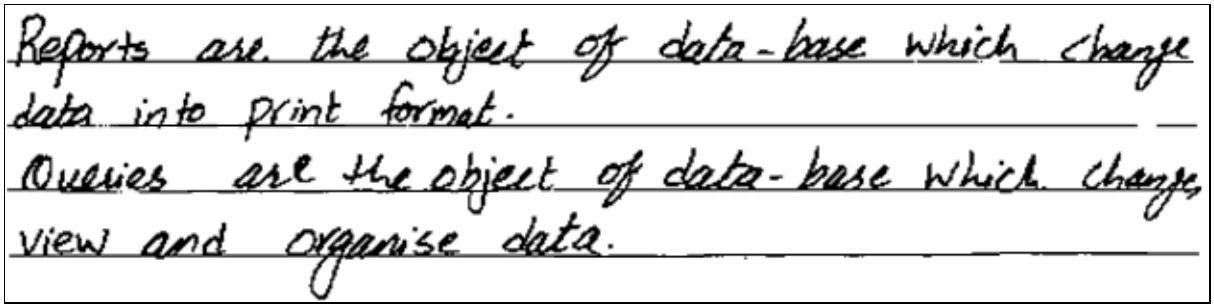
Better responses showed a good basic knowledge of database terminologies, i.e. reports and queries. These responses included: Reports display data from database in a formatted way/ reports are printed information from the database/ they provide summary of data/ they are presentable or readable format of data; Queries are used to extract data from a table or multiple tables according to specific criteria/ query is request for data from a database table/ query is used to manipulate data/ query provides a view of table on the basis of some specific criterion/ query allows user to filter data from database table according to their requirements.

Example:



Weaker responses showed lack of knowledge of basic terminologies of database. These responses failed to define reports and queries using proper keywords. These responses included: Reports are objects of database/ they change database into print format/ they combine all records/ they are combination of fields and records; Queries are an object of database/ they change view of data and organise it/ they are easy to use programming languages/ they make data simple as per need.

Example:



Question 2a:

Ali, a database administrator, wants to apply relationship among currently unrelated tables. The names of these tables are Orders, Items, ItemsOrders, Customers and Addresses.

Suggest the type of relationship that he should apply on the given set of tables.

i.

Orders			
Order_ID	**Customer_ID**	**Order_Date**	**Amount**
801	101	24/11/2018	14600
802	102	27/11/2018	9800

Items		
Item_ID	**Item_Name**	**Item_Description**
401	Mini Fridge	Eco friendly small-sized fridge.
402	Water Dispenser	Energy efficient, cool and hot water dispenser.
403	Rechargeable Fan	Portable and rechargeable DC fan.

ItemsOrders	
Order_ID	**Item_ID**
801	401
801	402
802	402
802	403

Type of Relationship:

_____

ii.

Customers		
Customer_ID	**Customer_Name**	**Address_ID**
101	Muhammad Ali	501
102	Anusha Mateen	502

Addresses	
Address_ID	**Customer_Address**
501	House No. 50, ABC Block 2, Karachi
502	House No. 80, DEF Block 3, Sukkur

Type of Relationship: _____

iii.

Customers		
Customer_ID	**Customer_Name**	**Address_ID**
101	Muhammad Ali	501
102	Anusha Mateen	502

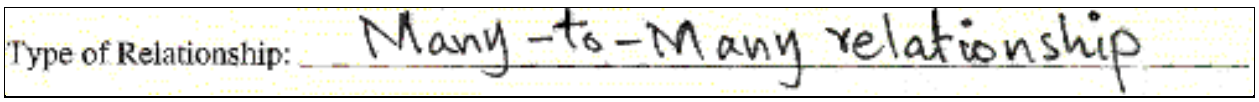
Orders			
Order_ID	**Customer_ID**	**Order_Date**	**Amount**
801	101	24/11/2018	14600
802	102	27/11/2018	9800
803	102	30/11/2018	6600
804	101	04/12/2018	3400

Type of Relationship: _____

(i)

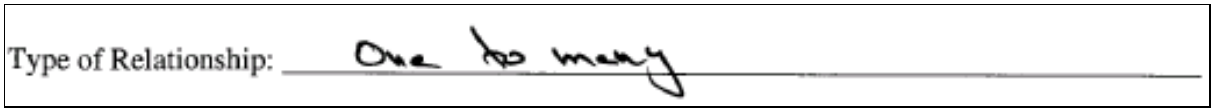
Better responses depicted that candidates successfully applied their knowledge of relationships in database and suggested correct name of relationship, i.e. many-to-many relationship.

Example:



Weaker responses demonstrated that candidates failed to suggest the correct type of relationship on the basis of given set of tables. Most of these responses suggested the one-to-many relationship. Similarly, some of these responses suggested either the many-to-one or the one-to-one relationship.

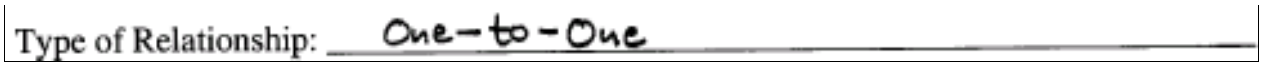
Example:



(ii)

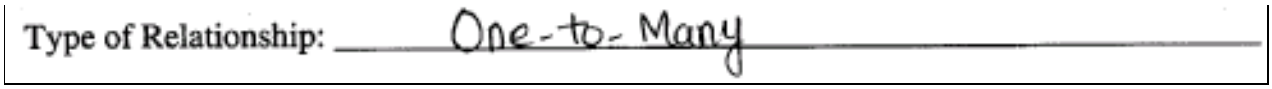
Better responses depicted that candidates successfully applied their knowledge of relationships in the database and suggested the correct name of the relationship, i.e. one-to-one relationship.

Example:



Weaker responses struggled to suggest the correct type of relationship on the basis of given set of tables. Most of these responses suggested the one-to-many relationship. Some of these responses suggested the many-to-many relationship.

Example:



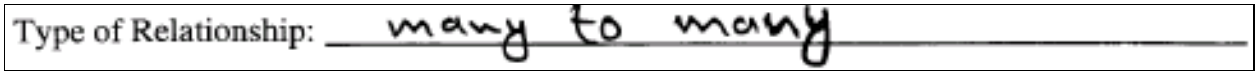
(iii)

Better responses depicted that candidates successfully applied their knowledge of relationships in the database and suggested the correct name of the relationship, i.e. one-to-many relationship or many-to-one relationship.

Example:

Weaker responses were unable to suggest the correct type of relationship on the basis of the given set of tables. Most of these responses suggested the one-to-one relationship. Some of these responses suggested the many-to-many relationship.

Example:



Question 2b:

Read the given database table.

Student_ID	Student_Name	City
101	Muhammad Ali	Karachi
102	Anusha Mateen	Sukkur
103	Shahid Khan	Sukkur

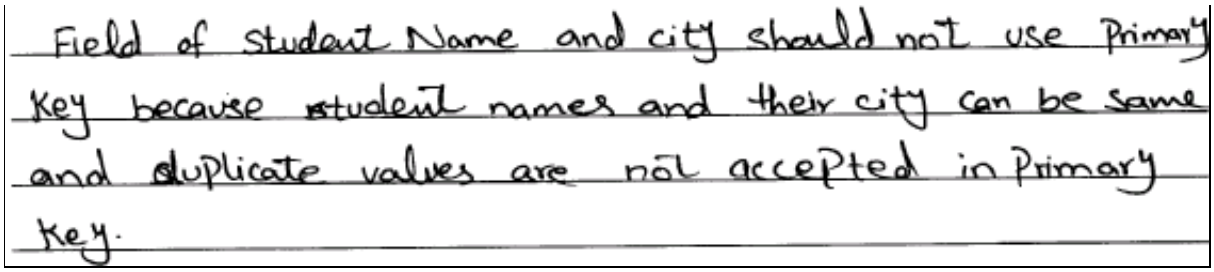
Which field should NOT be selected as primary key for this database table and why?

(**Note**: Most of the candidates attempted this question well.)

Better responses revealed that candidates had good conceptual knowledge about the field that cannot be selected as a primary key and why is that so. Most of these responses identified City as a field that cannot be selected as primary key.

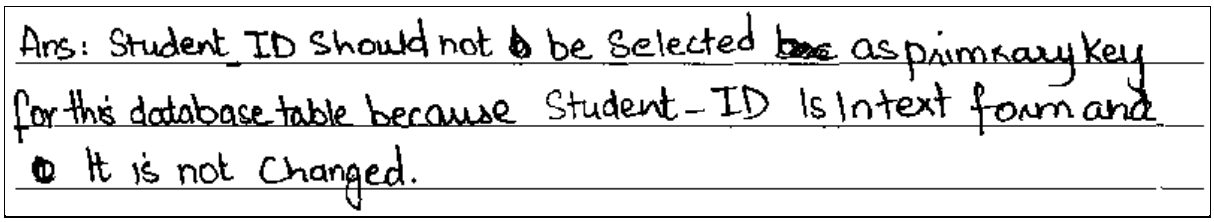
However, some of these responses mentioned both fields City and Student_Name that cannot be selected as primary key because usually duplication of names happens in large database tables. Therefore, they were awarded marks on it. Likewise, reason written by these responses for not selecting City and Student_Name as primary key was clear and common, i.e. primary key should be unique or these fields are not unique.

Example:


Field of Student Name and city should not use Primary Key because student names and their city can be same and duplicate values are not accepted in Primary Key.

Weaker responses identified Student_ID as a field that cannot be selected as primary key. The reasons written by these responses included: it is in text form; it is in numeric form; it is not changed; it is in normal form.

Example:


Ans: Student_ID should not be selected as primary key for this database table because Student-ID is in text form and it is not Changed.

Question 3a:

Read the given database table containing the data of employees.

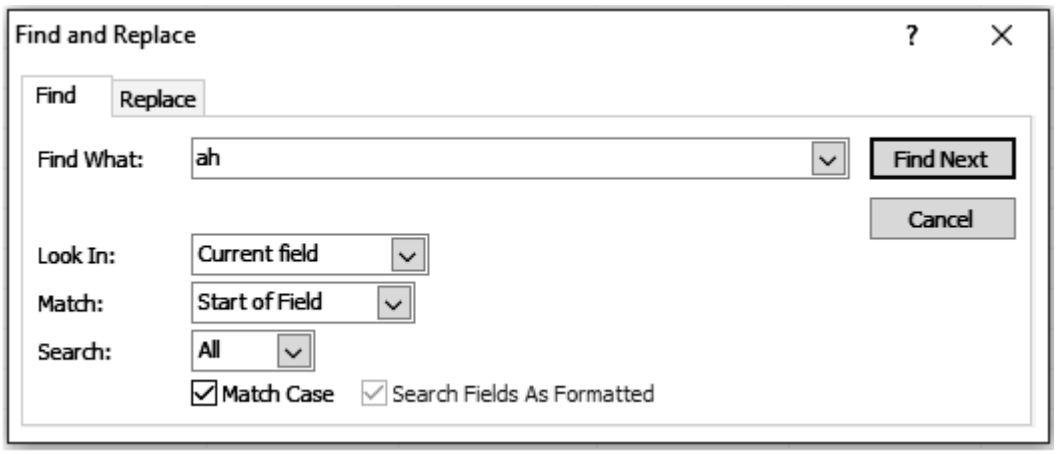
Employee_ID	First_Name	Last_Name	Age
E101	Fahad	Hussain	30
E102	Kashif	Khan	40
E103	Ahad	Zain	50
E104	Hammad	Ali	20

What result would be shown by MS Access in the First_Name field according to the options selected in the given Find and Replace dialogue box?

i.

Result Shown by MS Access: _____

ii.

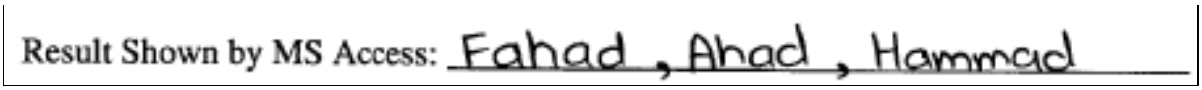


Result Shown by MS Access: _____

(**Note**: Most of the candidates did not perform well in this question.)

Better responses demonstrated strong concepts of searching in MS Access using Find and Replace dialogue box. These responses depicted that candidates paid attention to each criterion of this dialogue box and wrote correct results for both part (i) and part (ii).

Example (i):



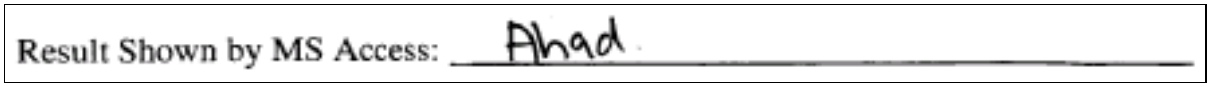
Example (ii):

Weaker responses depicted that candidates did not pay attention to the details shown in the Find and Replace dialogue box. These responses only focused on Find What field, i.e. 'ha' and did not focused on Match field, i.e. Any Part of Field. Due to this reason, they wrote Hammad instead of writing Fahad, Ahad, Hammad. Some of these candidates managed to answer part (i) of this question.

Likewise, these responses focused on Find What, i.e. 'ah' and did not focus on Match Case field. Due to this reason, they wrote Ahad instead of none or no result.

Example (i):

Example (ii):



Question 3b:

Read the given database table containing the data of employees.

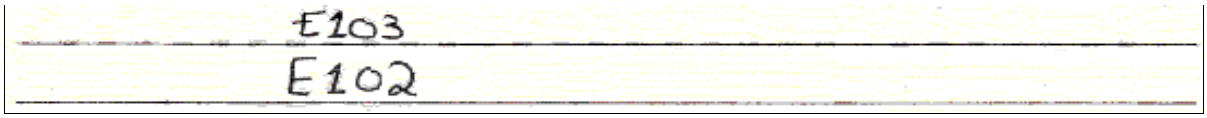
Employee_ID	First_Name	Last_Name	Age
E101	Fahad	Khan	30
E102	Kashif	Hussain	50
E103	Iqra	Hassan	40
E104	Hammad	Ali	20
E105	Aneela	Omar	35

i. If the ascending sort is applied on **Age** field of this table, then write the **Employee_ID** of record(s) that will appear below E105 in the correct sequence.

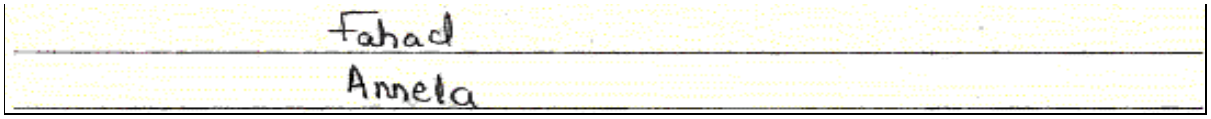
ii. If the descending sort is applied on **First_Name** field of this table, then write the **First_Name** of record(s) that will appear below the Hammad in the correct sequence.

Better responses revealed that candidates had a good conceptual knowledge of both ascending and descending sort and applied their knowledge of sorting correctly on the given scenario and wrote: E103 and E102 in part (i); Fahad and Aneela in part (ii).

Example (i):

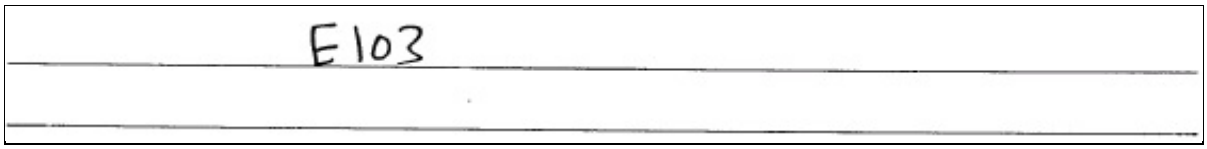


Example (ii):

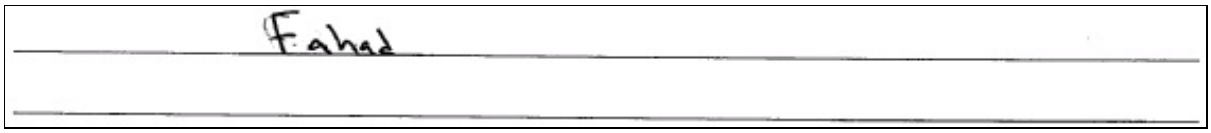


Weaker responses depicted that candidates did not understand the requirement of the question. They only wrote Employee_ID of one employee instead of two, i.e. E103 only rather than E103 and E102. Likewise, they only wrote First_Name of one employee instead of two, i.e. Fahad instead of Fahad and Aneela.

Example (i):



Example (ii):



Question 4:

A database table named Hotel_Info is setup to store information about hotels in a city. Few records from the table are shown below.

HID	Name of Hotel	Charges in PKR	Free Breakfast	Over All Rating	Distance from Airport in KM
DL150	Dreamland Hotel	5533	Yes	8	6
BM151	Blue Moon Hotel	3444	Yes	7.8	10
SV152	Sea View Hotel	8565	No	6.3	17
HF153	Hotel Farhan	3699	No	7.5	11
GR154	Green River Hotel	11000	Yes	6.6	5
WP155	White Palace Hotel	4477	No	5.4	18
RB156	Rainbow Hotel	6290	Yes	8.9	3

a. Show the output of the following query in design view of MS Access.

Field:	HID	Name of Hotel	Charges in PKR	Free Breakfast	Over All Rating
Table:	Hotel_Info	Hotel_Info	Hotel_Info	Hotel_Info	Hotel_Info
Sort:					
Show:	☐	☑	☑	☑	☐
Criteria:			>= 4000	= Yes	
or:					

Space for Output

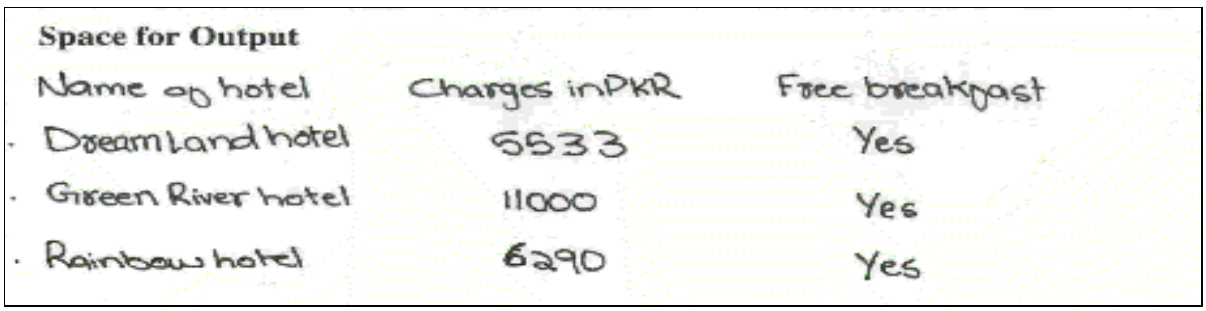
b. Complete the query in the given design view to select and show the **Name of Hotel** that are not far than a distance of 7 kilometres from the airport or have an **Overall Rating** of above 7.

Field:	HID	Name of Hotel	Charges in PKR	Overall Rating	Distance from Airport in KM
Table:	Hotel_Info	Hotel_Info	Hotel_Info	Hotel_Info	Hotel_Info
Sort:					
Show:	☐	☐	☐	☐	☐
Criteria:					
or:					

Better responses showed that candidates had a good understanding of query design view of MS Access and relational operators. These responses included the following records in the output:

Dreamland Hotel	5533	Yes
Green River Hotel	11000	Yes
Rainbow Hotel	6290	Yes

Example:



Weaker responses depicted that candidates perceived the >= as lesser than or equals to rather than greater than or equals to due to which they wrote Blue Moon Hotel and Hotel Farhan that has charges less than 4000. Moreover, some of these wrote all hotel records for output rather than writing the required records in output.

Example:

Space for Output

Blue Moon Hotel 3444 Yes
Hotel Farhan 3699 NO

Question 4b:

Better responses proved that candidates had good application skills of query design and completed the query given in the question correctly. These included: ticking the correct show box/ using appropriate relational operators/ using OR operator.

Example:

Field:	HID	Name of Hotel	Charges in PKR	Overall Rating	Distance from Airport in KM
Table:	Hotel_Info	Hotel_Info	Hotel_Info	Hotel_Info	Hotel_Info
Sort:					
Show:	☐	☑	☐	☐	☐
Criteria:					<= 7
or:				> 7	

Weaker responses revealed that candidates failed to complete the query in design view. Most of the responses marked tick on more than one Show boxes/ used AND operator (wrote both conditions in the same row) instead of OR operator/ used incorrect relational operator in query criteria, i.e. < 7 instead of <= 7

Example:

Field:	HID	Name of Hotel	Charges in PKR	Overall Rating	Distance from Airport in KM
Table:	Hotel_Info	Hotel_Info	Hotel_Info	Hotel_Info	Hotel_Info
Sort:					
Show:	☐	☑	☐	☑	☐
Criteria:		<=7		>=7	
or:					

Question 5:

The given table shows pairs of characteristics that differentiate between high and low level programming languages.

Identify the characteristic of high level programming languages from each pair by ticking (✓) the relevant box.

(**Note**: One high level programming language characteristic is identified from 1st pair of statements for your assistance.)

S.No.		
1	These programming languages are farther from human languages. ☐	These programming languages are closer to human languages. ☑
2	There is no need to have deep knowledge of hardware to write programs using these programming languages. ☐	Deep knowledge of hardware is required to write programs using these programming languages. ☐
3	These programming languages are used to write application programs. ☐	These programming languages are used to write hardware controlling programs. ☐
4	Programs that are written using these programming languages execute faster. ☐	Programs that are written using these programming languages execute slow. ☐
5	These programming languages provide many built-in functions. ☐	These programming languages do not provide the feature of built-in functions. ☐
6	It is difficult to trace errors in programs that are written using these programming languages. ☐	It is easy to trace errors in programs that are written using these programming languages. ☐

Better responses showed that candidates had good understanding of differentiation between high and low level programming languages. These responses ticked the relevant correct boxes for the purpose of differentiation.

Example:

S.No.		
1	These programming languages are farther from human languages. ☐	These programming languages are closer to human languages. ☑
2	There is no need to have deep knowledge of hardware to write programs using these programming languages. ☑	Deep knowledge of hardware is required to write programs using these programming languages. ☐
3	These programming languages are used to write application programs. ☑	These programming languages are used to write hardware controlling programs. ☐
4	Programs that are written using these programming languages execute faster. ☐	Programs that are written using these programming languages execute slow. ☑
5	These programming languages provide many built-in functions. ☑	These programming languages do not provide the feature of built-in functions. ☐
6	It is difficult to trace errors in programs that are written using these programming languages. ☐	It is easy to trace errors in programs that are written using these programming languages. ☑

Weaker response showed that candidates had a lack of understanding of the differences between high and low level programming languages. Most of them ticked incorrect boxes in the 3rd, 4th and 5th pair of characteristics.

Example:

S.No.		
1	These programming languages are farther from human languages. ☐	These programming languages are closer to human languages. ☑
2	There is no need to have deep knowledge of hardware to write programs using these programming languages. ☑	Deep knowledge of hardware is required to write programs using these programming languages. ☐
3	These programming languages are used to write application programs. ☐	These programming languages are used to write hardware controlling programs. ☑
4	Programs that are written using these programming languages execute faster. ☑	Programs that are written using these programming languages execute slow. ☐
5	These programming languages provide many built-in functions. ☐	These programming languages do not provide the feature of built-in functions. ☑
6	It is difficult to trace errors in programs that are written using these programming languages. ☐	It is easy to trace errors in programs that are written using these programming languages. ☑

Question 6a:

A Computer Science student writes steps of compilation process. Three of these steps are incorrect and given below. Each of these steps has exactly ONE incorrect word.

Identify the incorrect word in each step by underlining it and write the correct word.

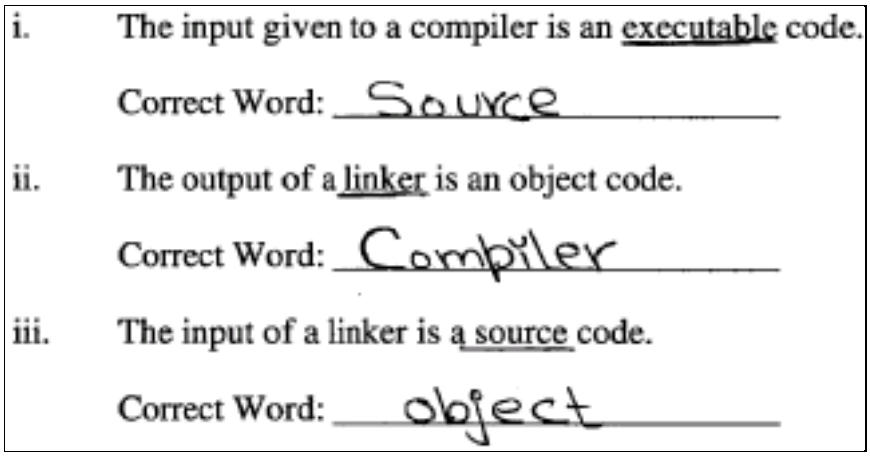
i. The input given to a compiler is an executable code.
Correct Word: _____

ii. The output of a linker is an object code.
Correct Word: _____

iii. The input of a linker is a source code.
Correct Word: _____

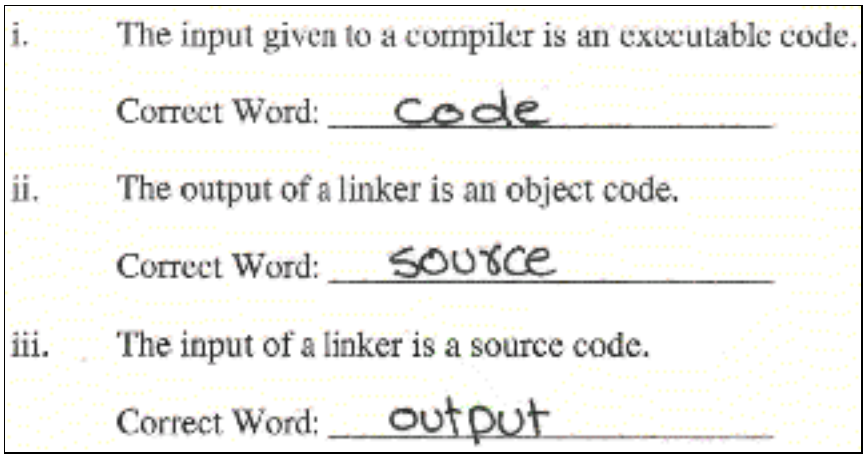
Better responses depicted that candidates had a good understanding of the compilation process of a C program. They underlined each incorrect term in each step and replaced it with the correct term. They underlined executable in first step and replaced it with source. Likewise, they underlined either linker or object in the second step and replaced them with compiler or executable respectively. Similarly, they underlined linker or source in third step and replaced them with compiler or object respectively.

Example:



Weaker responses proved that candidates had a poor understanding of the compilation process of a C program. One of the glaring mistakes done by them was that they did not underline the incorrect terms. Moreover, most of them wrote the terms that are given in the compilation process steps without checking that whether they make any connection with the given steps or not.

Example:



Question 6b:

Complete the given C program that prints Computer Science as shown in the given image.

```
Computer
Science
```

(**Note:** You must use escape sequence in your program to print Science on the next line.)

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

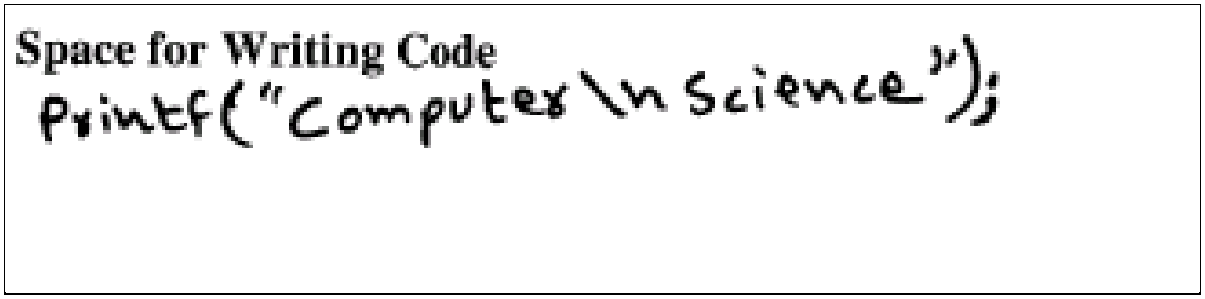
Space for Writing Code

```
return 0;
}
```

(**Note:** This question was well-attempted by the candidates.)

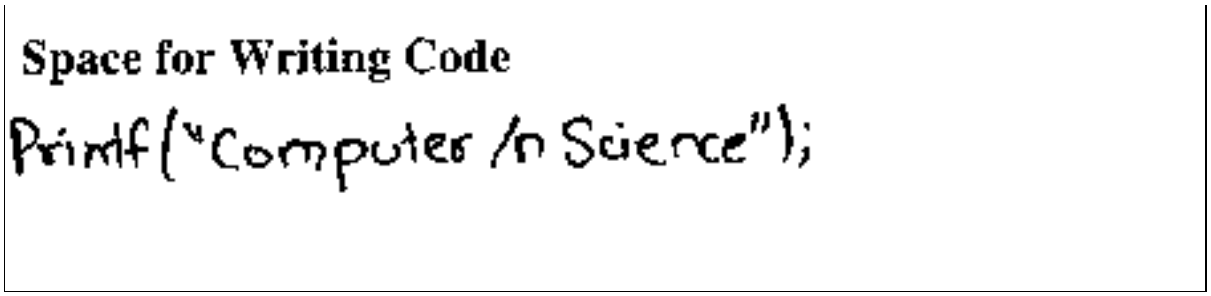
Better responses demonstrated that candidates had good basic application skill of C programming. They wrote the correct syntax of printf function along with the message to print and used \n (escape sequence) at the right place of the program to print Computer at the first line and Science at the second line.

Example:



Weaker responses revealed that most of the candidates managed to write the correct syntax of printf program to output Computer Science on the screen but they failed to use the escape sequence correctly to output Computer at the first line and Science at the second line.

Example:



Question 7:

Read the given C program.

```
#include <stdio.h>
int main()
{
   int c, n, t, r, x = 0;
   scanf("%d", &n);
   for(c = 1; c <= n; c++)
   {
      x = c * 9;
      r = x % 5;
      t = t + r;
   }
   printf("Result = %d", t);
   return 0;
}
```

a. If the input value in this program is **4** and **for** loop is executed, then complete the given table by writing the missing values of **x**, **r** and **t** for each iteration (repetition) of this program.

Iteration	x	r	t
1st			
2nd			
3rd			
4th			

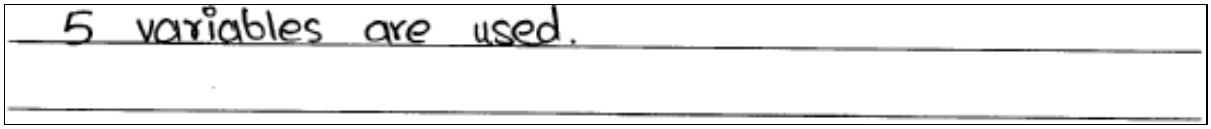
b. How many variables are used in this program?

Better responses proved that candidates had good application skills of various concepts of C programming that included: For Loop; arithmetic operations; use of modulus operator; totaling process; declaring variables.

Example (a):

Iteration	x	r	t
1st	9	4	4
2nd	18	3	7
3rd	27	2	9
4th	36	1	10

Example (b):

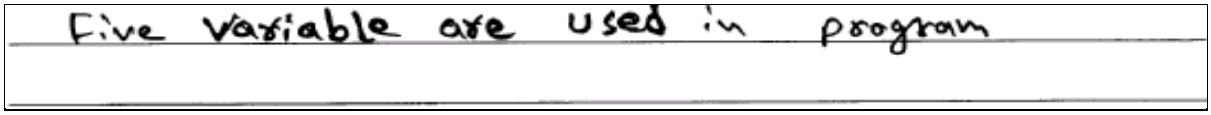


Weaker responses showed that candidates had lack of application skills required to interpret the program given in the question and complete the table with correct values in each iteration. Most of them wrote the values of variable **x** correctly but did not write correct values for **r** (divided x by 5 instead of applying modulus operation) and **t** variables. Some of them managed to answer part of **(b)** of this question, i.e. 5 variables, but most of them wrote 4 variables that might be due to the reason that candidates did not pay a detailed attention to the program given in the question and counted only those variables that are inside the loop, i.e. **x**, **c**, **r** and **t**, and ignored the variable **n** that stored the input value.

Example (a):

Iteration	x	r	t
1st	9	1.8	1.8
2nd	18	3.6	5.4
3rd	27	5.4	10.8
4th	36	7.2	18

Example (b):



Question 8:

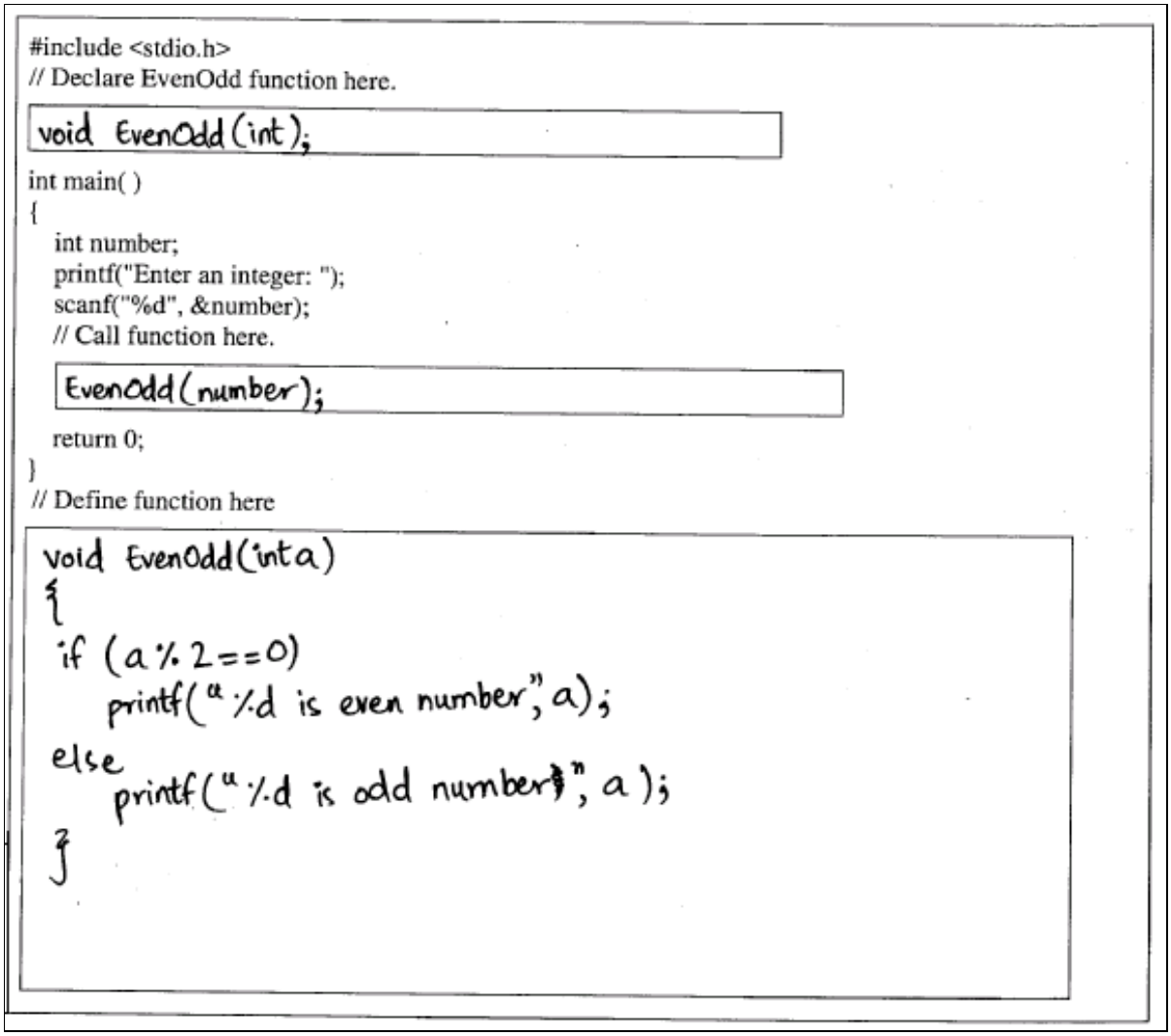
Complete the given code for a user-defined function named 'EvenOdd' that would determine whether an integer is even or odd.

(**Note**: An even number is divisible by 2 while an odd number is not divisible by 2.)

```
#include <stdio.h>
// Declare EvenOdd function here.
[                                        ]
int main( )
{
   int number;
   printf("Enter an integer: ");
   scanf("%d", &number);
   // Call function here.
   [                                     ]
   return 0;
}
 // Define function here
[                                        ]
[                                        ]
[                                        ]
[                                        ]
[                                        ]
```

Better responses showed that candidates had strong application skills of the user-defined functions in C programming. The code written by them included: correct syntax to declare the function; to call the function; to define the function; to check if a number is even or odd by finding modulus of number with 2; to print appropriate messages for even and odd numbers.

Example:



```
#include <stdio.h>
// Declare EvenOdd function here.
void EvenOdd(int);
int main( )
{
   int number;
   printf("Enter an integer: ");
   scanf("%d", &number);
   // Call function here.
   EvenOdd(number);
   return 0;
}
// Define function here
void EvenOdd(int a)
{
 if (a % 2==0)
    printf("%d is even number", a);
 else
    printf("%d is odd number", a);
}
```

Weaker responses proved that candidates did not have good application skills of user-defined functions in C programming. The code written by these responses included: incorrect function name; incorrect function arguments or parameters; using numeric values in function call; incorrect syntax to define a function; incorrect IF condition or using loops instead of IF condition to check whether an input value is even or odd; incorrect syntax to print messages for even and odd numbers.

Example:

```
#include <stdio.h>
// Declare EvenOdd function here.
int evenodd (evenodd)
int main( )
{
   int number;
   printf("Enter an integer: ");
   scanf("%d", &number);
   // Call function here.
         (evenodd)
   return 0;
}
 // Define function here
int evenodd (evenodd)
 {
  if (number/2);
  Printf (number is even);
  else
printf(number is odd);
   }
```

Question 9:

Read the given C language code.

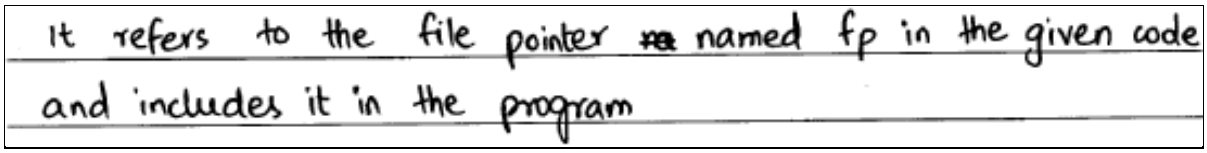
```
#include <stdio.h>
#include <stdlib.h>
int main()
{
  int n;
  FILE *fp;
  fp = fopen("C:\\data.txt","w");
  if(fp == NULL)
  {
    printf("Error");
    exit(1);
  }
  printf("Enter number: ");
  scanf("%d",&n);
  fprintf(fp,"%d",n);
  fclose(fp);
  return 0;
}
```

Interpret the purpose of the code lines given from this code.

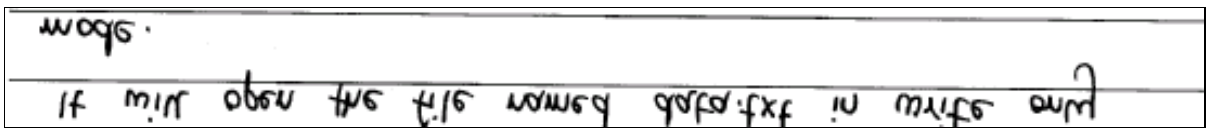
i. FILE *fp;
ii. fp = fopen("C:\\data.txt","w");
iii. fprintf(fp,"%d",n);
iv. fclose(fp);

Better responses depicted that candidates had good application skills of file handling in C programming and interpreted the given codes correctly. The interpretations written by them included: (a) declares the file pointer fp/ defines the structure of the file/ declares the file structure/ creates the file pointer; (b) opens a file called data.txt in write mode/ opens a file named data.txt in write mode; (c) writes (store) value of variable n to the fb file pointer/ copies the value of n in file pointer; (d) closes the file pointer fp;

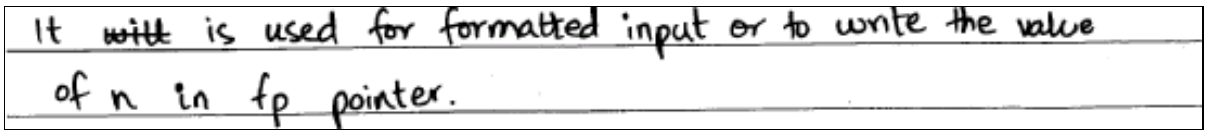
Example (a):



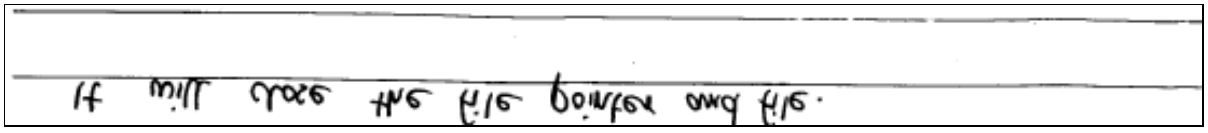
Example (b):



Example (c):



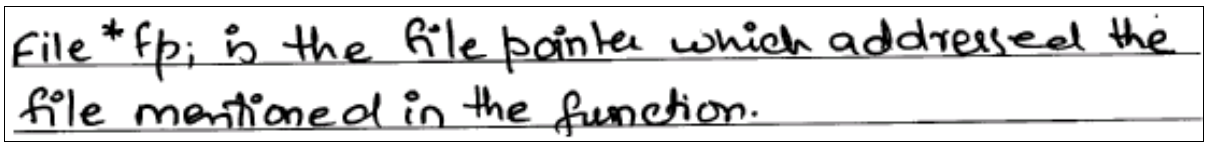
Example (d):



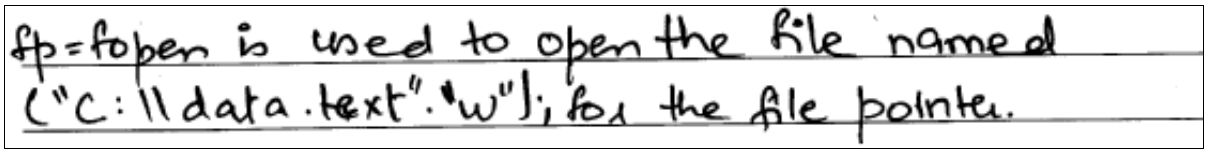
Weaker responses showed that candidates failed to interpret the codes given in the question. They mostly wrote with reference of file instead of file pointer. The interpretations written by them included: (a) pointer goes on file/ to show files (b) to open the file and read it (c) to print the file/ to show results/ to print the data in file (d) to close the file/ to close the program.

Most of these responses managed to interpret the code given in part (d).

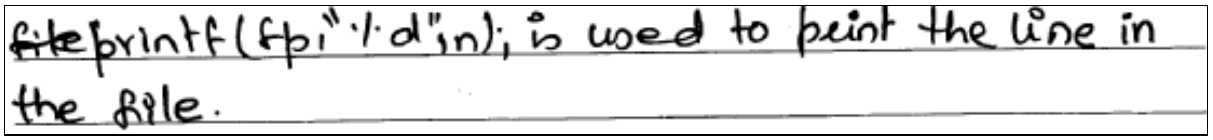
Example (a):



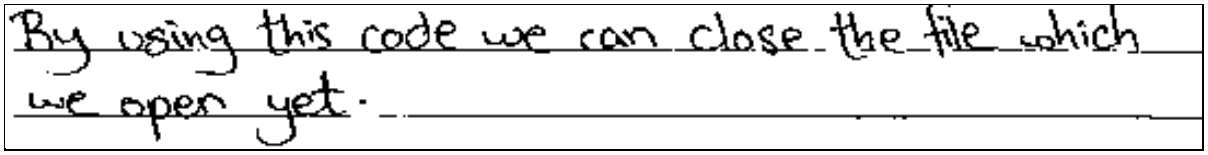
Example (b):



Example (c):



Example (d):



Extended Response Questions (ERQs)

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

Question 10a:

A gymnasium trainer wants to calculate the maximum, minimum and the average weight of the gymnasium (gym) members.

Write a program using C language that

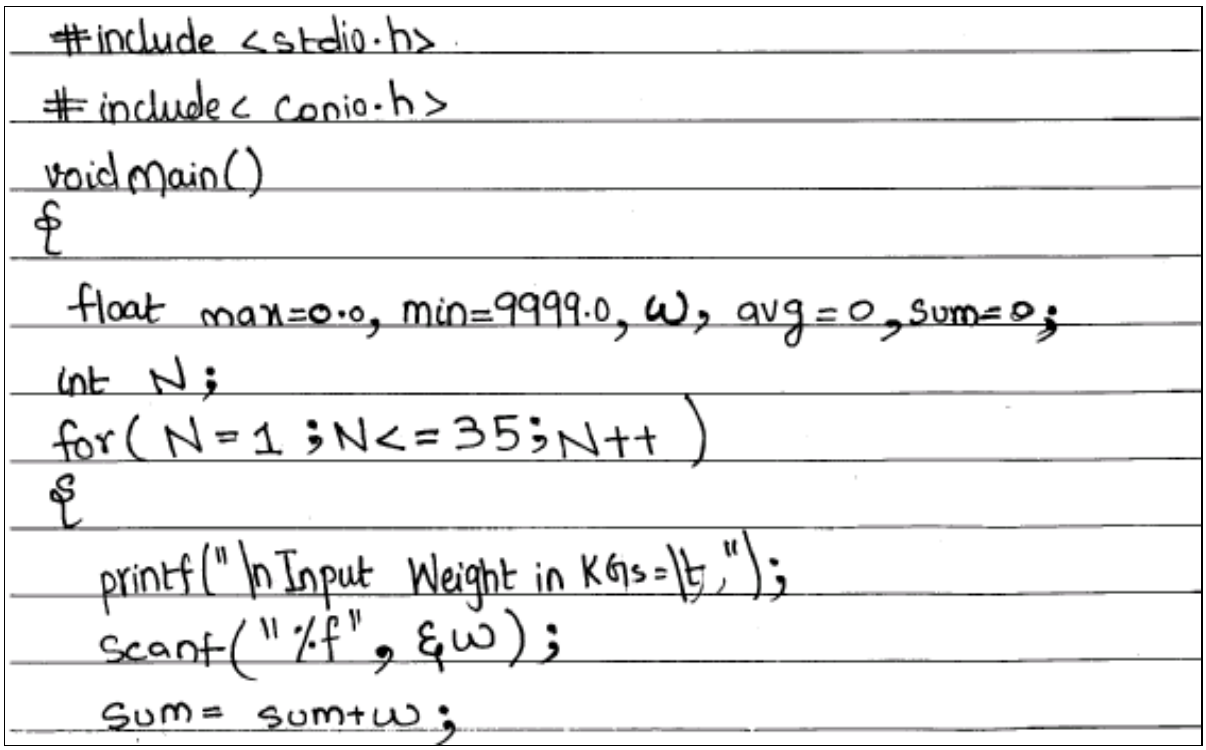
- inputs weight of 35 gym members.
- calculates minimum, maximum and average weight values.
- outputs minimum, maximum and average weight values.

(**Note**: Use the For loop for repeating the code.)

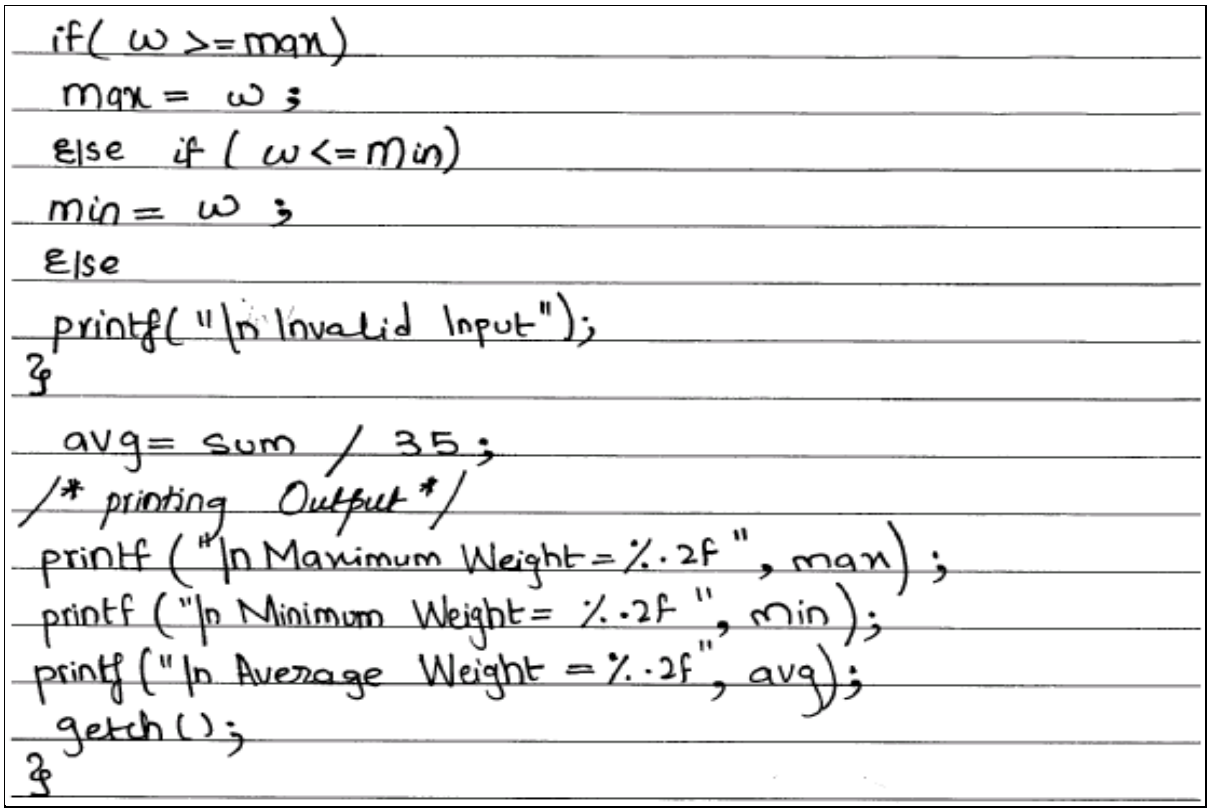
(**Note:** Most of the students attempted part (a) and they did well in it.)

Better responses showed that candidates had a good understanding of various concepts of the C programming that are used to write this programming code. The programs written by them included: declaration of variables; definition of variables; For loop with initial and final values; input and output; totaling, calculating minimum and maximum weight values. Most of the candidates used array in the solution of this problem.

Example:



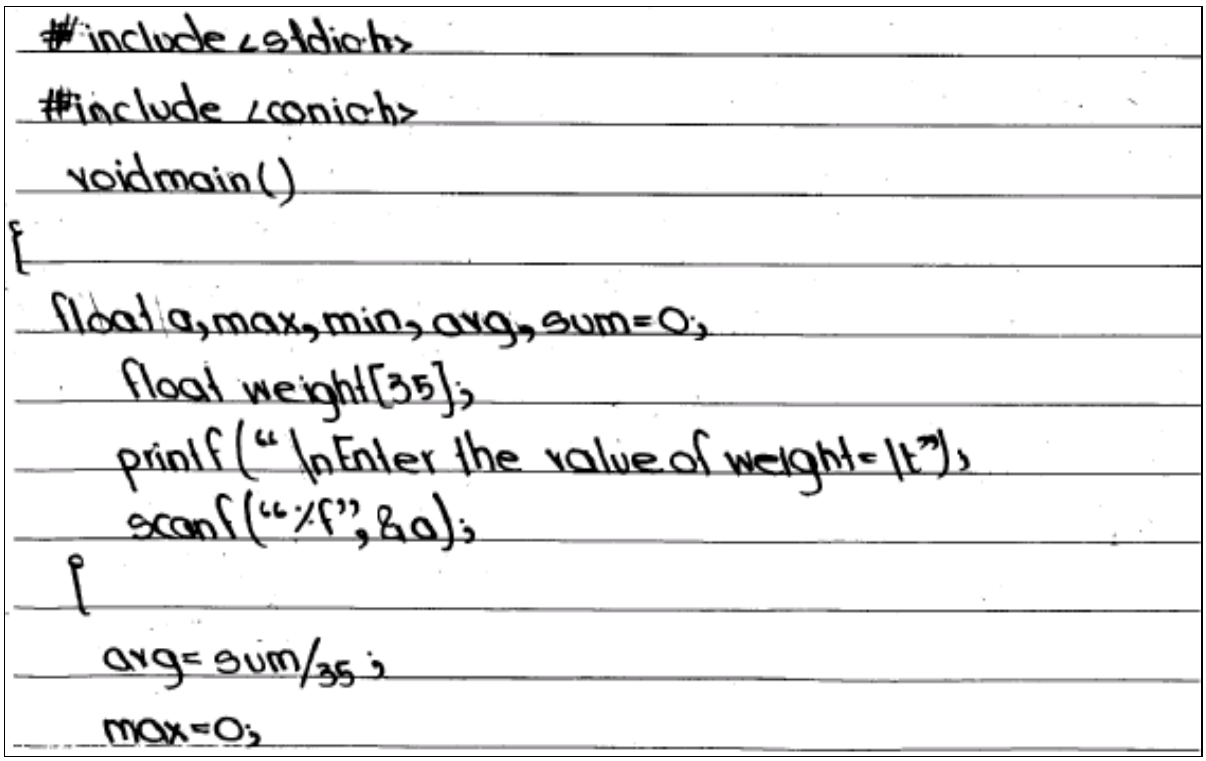
```
#include <stdio.h>
#include< conio.h>
void main()
{
 float max=0.0, min=9999.0, w, avg=0, sum=0;
int N;
for( N=1 ;N<=35;N++ )
{
   printf("\nInput Weight in KGs=\t,");
   scanf("%f", &w);
   sum= sum+w;
```



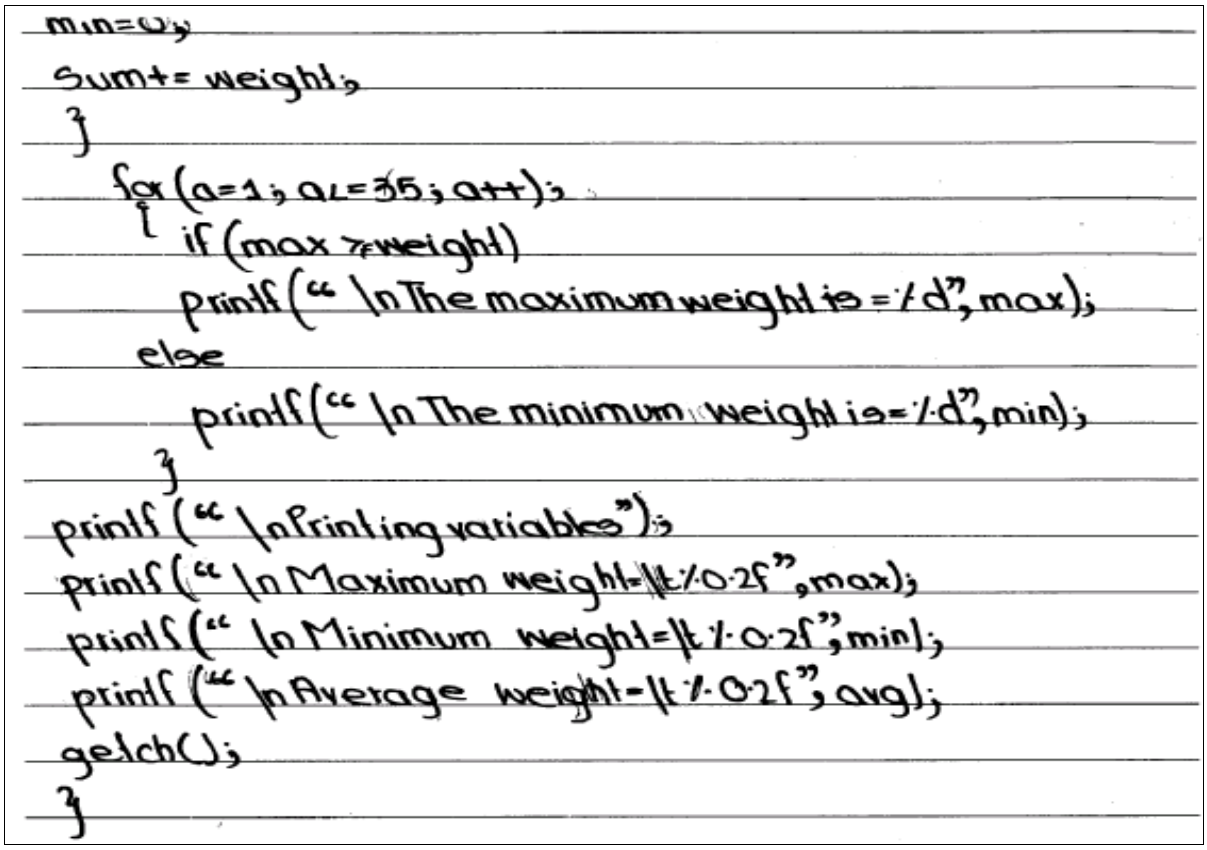
```
if( w >= max)
max = w ;
else if ( w <= min)
min = w ;
else
printf("\n Invalid Input");
}
avg= sum / 35;
/* printing Output */
printf ("\n Maximum Weight = %.2f ", max) ;
printf ("\n Minimum Weight = %.2f ", min);
printf ("\n Average Weight = %.2f", avg);
getch();
}
```

Weaker responses demonstrated that candidates managed to write programs that included few correct programming concepts such as: input and output of values; using loop statement; calculation of average. However, these candidates failed write the C programming logic to calculate minimum and maximum weight values and declare all the required variables with initial values.

Example:



```
#include<stdio.h>
#include<conio.h>
void main()
{
float a,max,min,avg,sum=0;
   float weight[35];
   printf("\nEnter the value of weight=\t");
   scanf("%f",&a);
{
   avg=sum/35;
   max=0;
```



```
min=0;
Sum+= weight;
}
for (a=1; a<=35; a++);
{
  if (max >weight)
    printf("\nThe maximum weight is =%d", max);
  else
    printf("\nThe minimum weight is=%d", min);
}
printf("\nPrinting variables");
printf("\nMaximum weight=%t%0.2f", max);
printf("\nMinimum weight=%t%0.2f", min);
printf("\nAverage weight=%t%0.2f", avg);
getch();
}
```

Question 10b:

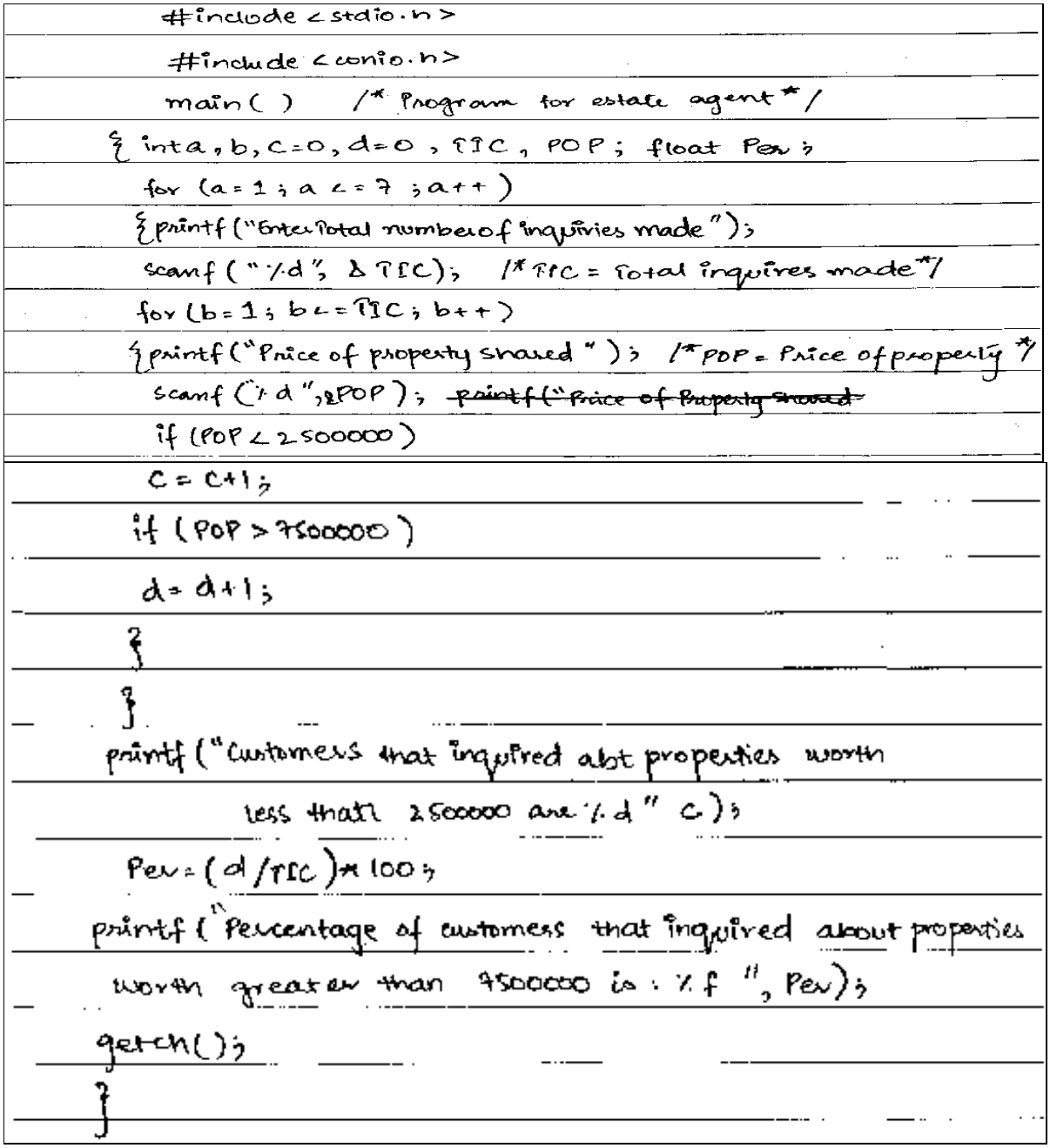
Adil is an owner of a real estate agency. He weekly advertises about the houses for sale in his area. He needs a software to efficiently manage his real estate agency work.

Write a program using C language that:

- inputs the total number of inquiries made by customers on a daily basis in a week.
- inputs the price of properties shared with the customers daily.
- calculates and outputs the number of customers that inquired about the properties worth less than Rs 2,500,000.
- calculates and outputs the percentage of customers that inquired about the properties worth greater than Rs 7,500,000.

Better responses showed that candidates had applied programming concepts on the given scenario correctly. The programs written by them included: declaration and definition of variables; For loop with initial and final values to input number of enquiries each day; input; calculate and output total enquiries with price lesser than 2,500,000; For loop to enter the cost of house told in each enquiry; calculate and output percentage of enquiries with price greater than 7,500,000.

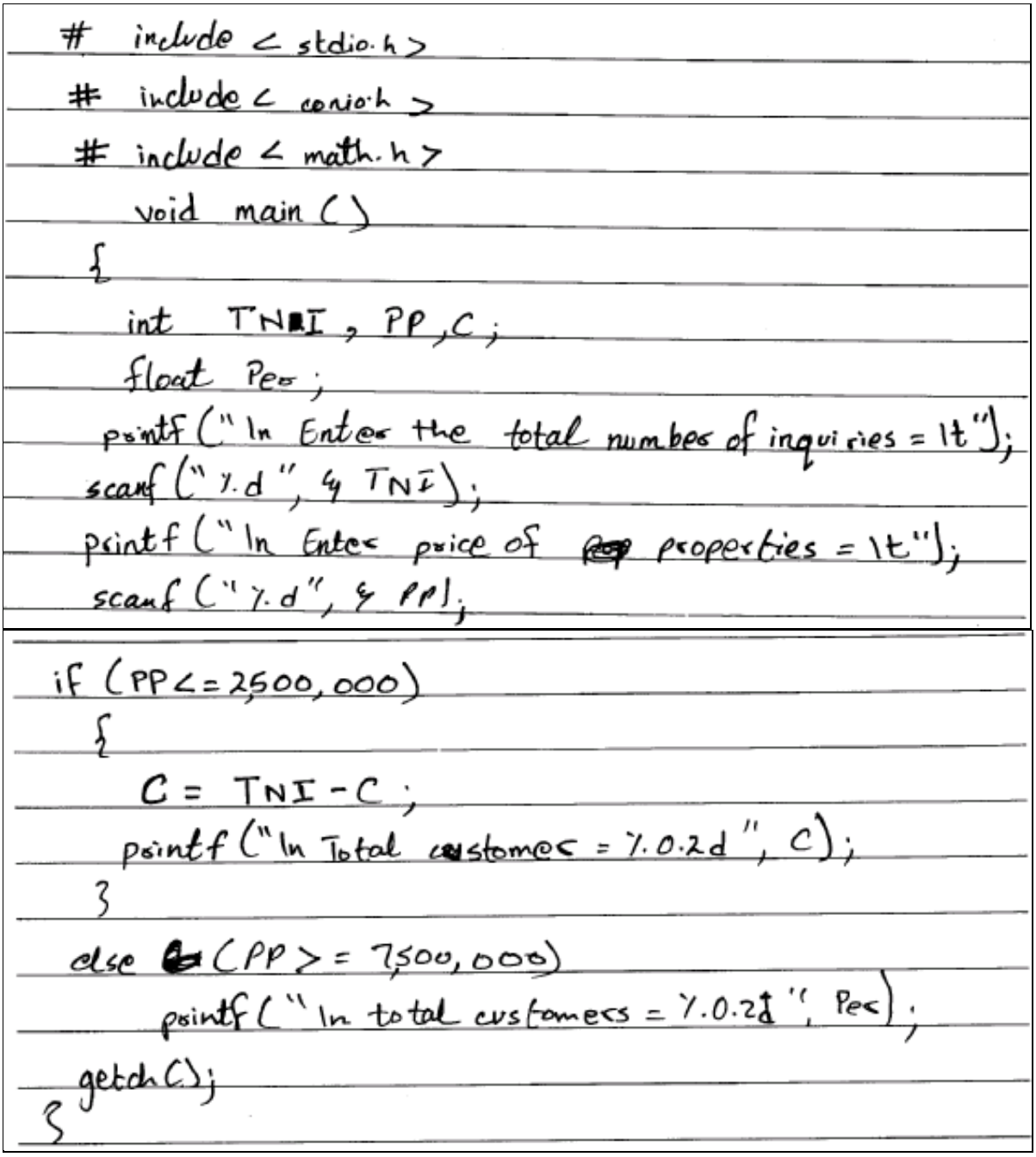
Example:



```
#include <stdio.h>
#include <conio.h>
main( )   /* Program for estate agent */
{ int a, b, c=0, d=0, TIC, POP; float Per;
for (a=1; a<=7; a++)
{printf("Enter Total number of inquiries made");
scanf("%d", &TIC);   /* TIC = Total inquires made */
for (b=1; b<=TIC; b++)
{printf("Price of property shared");  /* POP = Price of property */
scanf("%d", &POP);  printf("Price of Property shared"
if (POP<2500000)
c = c+1;
if (POP>7500000)
d = d+1;
}
}
printf("Customers that inquired abt properties worth
less than 2500000 are %d" c);
Per = (d/TIC)*100;
printf("Percentage of customers that inquired about properties
worth greater than 7500000 is: %f", Per);
getch();
}
```

Weaker responses showed that candidates failed to construct the logic in their programs to solve the give problem. They managed to write the correct code to input and output values and calculate the average. They were not able to correctly write the programming code for concepts that included: For loop for daily enquires; calculating total number of enquiries per day; For loop to input the value of houses in daily enquiries; counting the houses with value more than 7500000 and less than 2500000.

Example:



```
# include < stdio.h >
# include < conio.h >
# include < math.h >
   void main ( )
{
   int TNI , PP , C ;
   float Per ;
   printf (" \n Enter the total numbers of inquiries = \t ");
   scanf (" %d ", & TNI );
   printf (" \n Enter price of properties = \t ");
   scanf (" %d ", & PP );

if (PP <= 2,500,000)
   {
      C = TNI - C ;
      printf (" \n Total customer = %0.2d ", C );
   }
else (PP >= 7,500,000)
      printf (" \n total customers = %0.2f ", Per );
   getch ( );
}
```

Note: Less than 2% of the candidates attempted the Visual Basic (VB) programming section. It was good to see that candidates tried to attempt the VB Programming Section. However, candidates were unable to score satisfactory marks due to the lack of practice. To improve the score in this section, candidates and teachers are advised to go through the SLOs of VB programming 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.