Aga Khan University Examination Board

Notes from E-Marking Centre on HSSC-II Computer Science Examination May 2018

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 2018 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 definitions of basic terminologies of the database, data types, primary key, sorting, query in design view, types of programming languages, conversion of arithmetic expression into C program, building logic using selection statement were well attempted. However, questions based on values returned by main function and arguments passed to the main function, determining the output and purpose of programs, user-defined functions in C programming, file management in C programming and visual basic programming questions were generally not well attempted.

Detailed Comments:

Constructed Response Questions (CRQs)

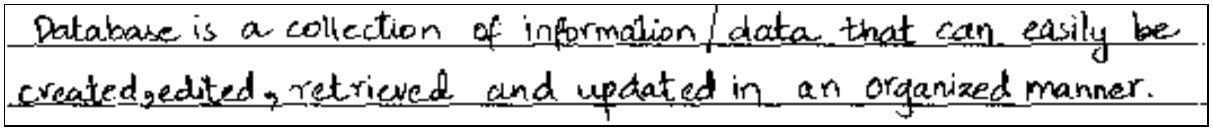
Question 1:

Define the following terms.

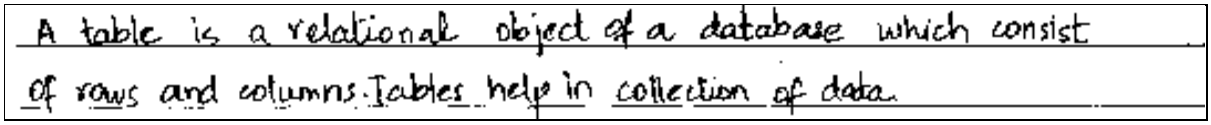
i. Database
ii. Table
iii. Query
iv. Field
v. Record

Better responses depicted that candidates had good knowledge about the basic terminologies of database and defined the terms given in this question correctly such as a database is a collection of organized information/ a table consists of rows and columns/ a query is used to retrieve data from the database on the basis of a specific criteria/ a database field is a single piece of information from a record/ records are rows in a database table and composed of fields.

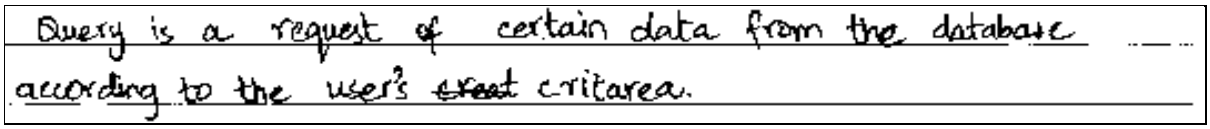
Example (i):


Database is a collection of information / data that can easily be created, edited, retrieved and updated in an organized manner.

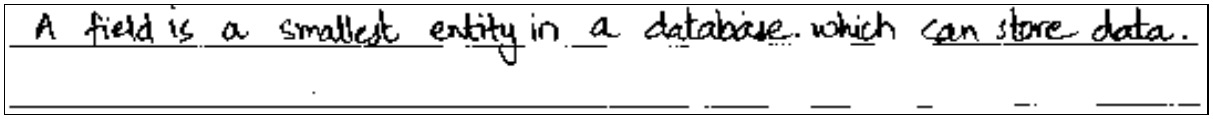
Example (ii):


A table is a relational object of a database which consist of rows and columns. Tables help in collection of data.

Example (iii):


Query is a request of certain data from the database according to the user's ~~exact~~ criteria.

Example (iv):


A field is a smallest entity in a database which can store data.

Example (v):

A record is the information present in the entire row about a specific person, company or etc. Record is a collection of fields.

Weaker responses failed to define the terms given in this question correctly which means candidates did not know the basic terminologies of the database. For example, the database is a program/ all records have the table/ query connects tables to each other/ field is a row in table/ record is a column in a table. A few of these responses managed to define table and/ or record with reference to the database.

Example (i):

Database is set of program one computer to other. Database is different form, report

Example (ii):

Table is made in data sheet view. It is used in all Record

Example (iii):

Query is specific constract to one or more table added the table

Example (iv):

Field is used to data type of name field is a row in table. Field is used to all data of entity

Example (v):

Record is a column in table. Record is used to data of attributes in table

Question 2:

Fatima has created a database table named **Product**. Few records of this table are shown below.

Prod_ID	Name	Expire_Date	Imported	Price per Item
P101	Natural Milk	10/09/2018	☑	125
P102	Frozen Kebab	09/15/2018	☐	750
P103	Yogurt	01/26/2019	☑	140
P104	Frozen Fish	10/09/2018	☐	750
P105	Frozen Kebab	02/20/2019	☑	610

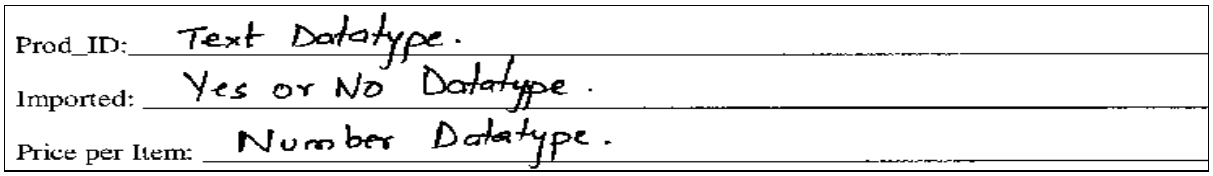
a. Write the data types she would have selected for the fields given below.

- Prod_ID
- Imported
- Price per Item

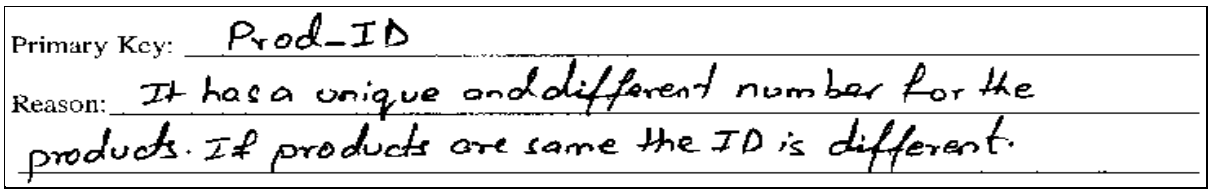
b. Write the name of the field from the given table which should be selected as a primary key. Give a valid reason to support your answer.

Better responses demonstrated that candidates practiced the creation of tables and selection of appropriate data types in MS Access such as text or alphanumeric for Prod_ID/ logical data type or yes or no for Imported/ currency or number. Moreover, these responses showed that candidates were able to choose the correct primary key for the database table given in this question, i.e. Prod_ID. Likewise, these responses wrote the correct reason to select Prod_ID as the primary key. For example, it uniquely identifies a record in a table.

Example (a):

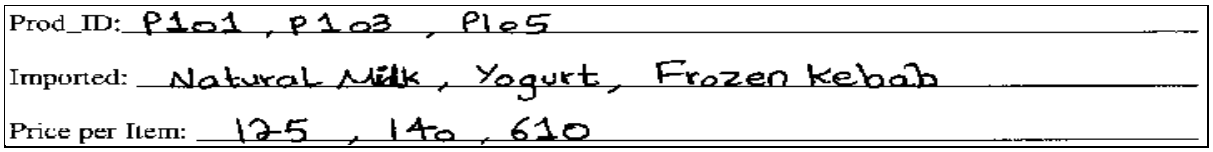


Example (b):

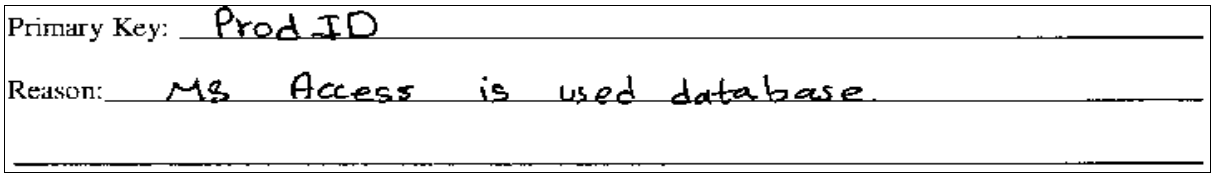


Weaker responses depicted the lack of practice of the creation of tables and selection of appropriate data types in MS Access. For example, these responses wrote data given in the table instead of the data type/ selected number or auto-number as the data type for Prod_ID/ selected OLE data type for Imported etc. However, most of these responses managed to identify the primary key from the table given in this question.

Example (a):



Example (b):



Question 3:

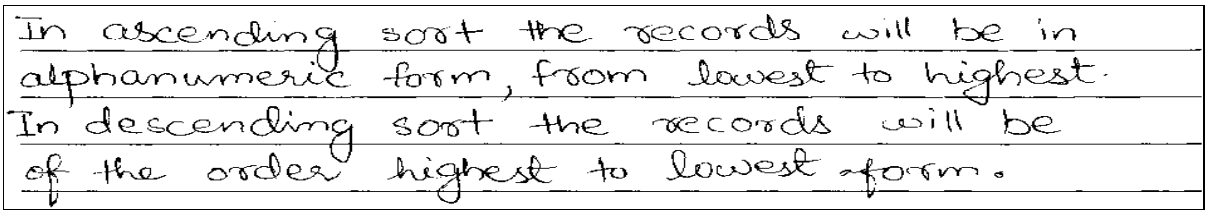
a. Describe ascending and descending sort.

b. Apply ascending sort on the given sequence of numbers. Show your working.
5^2, 1^9, 2^5, 4^2, 3^3

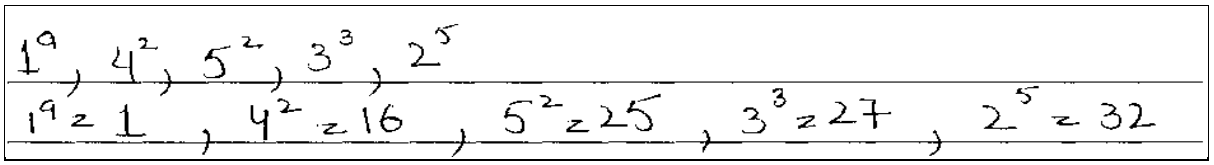
Better responses showed that candidates had a good conceptual understanding of ascending and descending sort. They wrote such as ascending sort arrange values from low to high/ descending sort arrange values from high to low.

Moreover, candidates were able to apply their understanding about ascending sort on the data given in part (b) of this question and sorted the data correctly, i.e. 1^9, 4^2, 5^2, 3^3, 2^5.

Example (a):

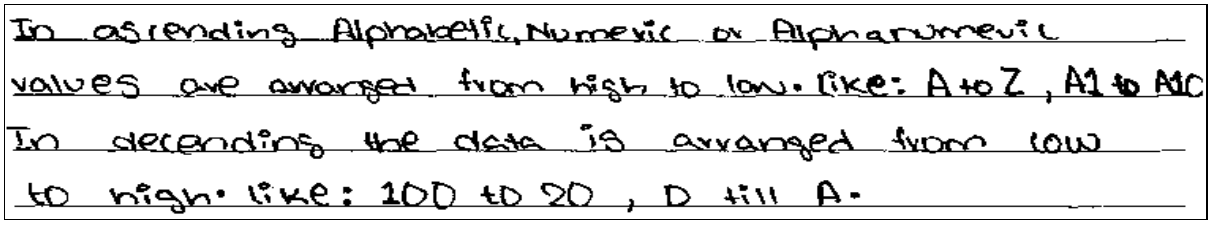


Example (b):

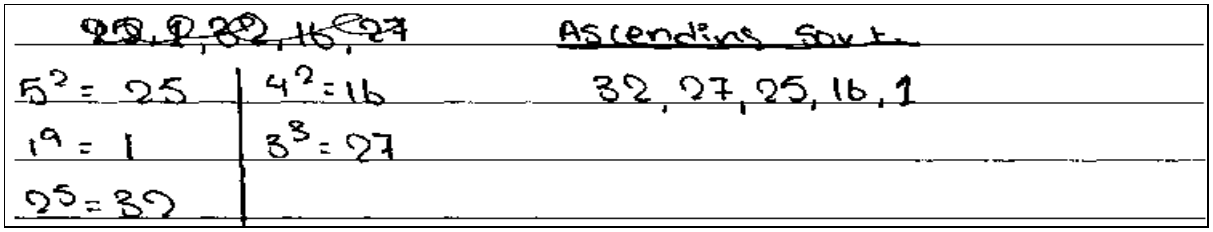


Weaker responses depicted that candidates mixed the concept of ascending and descending sort and wrote such as ascending sort arranges values from high to low and descending sort arranges values of low to high. Due to weaker concepts of ascending sort, candidates failed to apply it on the data given in part (b) of this question and sort the data correctly.

Example (a):



Example (b):



Question 4:

A database is set-up to store information about some of the high-rise buildings in Pakistan. Few records from the database are shown below.

Ref No.	Name of Building	City	Year	No. of Floors	Height (m)	Height (ft)
BIT100	Icon Tower	Karachi	2017	68	300	984
TO101	Riveria Mall	Lahore	2017	34	130	427
OT102	Sea Towers	Karachi	2012	30	120	394
CPT103	Centre Point Mall	Karachi	2013	28	117	384
TT104	Communication Tower	Islamabad	2011	25	113	371
TC105	The Centre Tower	Islamabad	2013	23	110	361
ASTP106	Information Technology Park	Lahore	2009	17	106	348

(**Note:** This is fictitious data about high-rise buildings in Pakistan.)

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

Field:	Ref No.	Name of Building	Year	Height (m)	Height (ft)
Table:					
Sort:					
Show:	☐	✓	✓	✓	☐
Criteria:			<=2013	>110	
or:					

Space for Output

b. Complete the query in design view below to select and show the Ref No., Name of Building, City, No. of Floors and Height (ft) of all buildings in Karachi or Lahore with height less than 450 feet.

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

Better responses exhibited that candidates had practiced queries in design view of MS Access. Such responses applied their concepts of query design on the given scenario to determine the output, i.e. Sea Towers 2012 120/ Center Point Mall 2013 117/ Communication Tower 2011 113. Moreover, these responses constructed the query in design view to show the data as per the criteria given in the question.

Example (a):

Space for Output

Name of Building	Year	Height
Sea Towers	2012	120
Centre Point Mall	2013	117
Communication Tower	2011	113

Example (b):

Field:	Ref No.	Name of Building	City	No. of Floors	Height (ft)
Table:					
Sort:					
Show:	☑	☑	☑	☑	☑
Criteria:			"Karachi"		< 450
or:			"Lahore"		

Weaker responses depicted that candidates had lack of conceptual knowledge to determine the output of a query in design view and wrote either irrelevant records or all records in the output such as Icon Tower 2017 300/ Riveria Mall 2017 130/ Information Technology Park 2009 106. However, most of such responses completed the query in design view and generally secured 2 or 3 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 they failed to apply the required criteria.

Example (a):

Space for Output

Name of building	Year	Height(m)
Icon Tower	2012	300
Riveria Mall	2013	130
Sea Towers	2011	120
Centre point Mall	2013	117
Communication Tower	2009	113
The Centre Tower		
Information Technology Park		

Example (b):

Field:	Ref No.	Name of Building	City	No. of floors	Height (ft)
Table:					
Sort:					
Show:	☑	☑	☑	☑	☑
Criteria:					<450
or:					

Question 5a:

Describe ONE main characteristic for each type of given programming language which makes it different from the other two.

Machine Language	Assembly Language	High-Level Language

Better responses showed good conceptual knowledge about the types of programming languages and described one main characteristic of each type of programming language that makes it different from the other two such as machine language is in the form of 1s and 0s and it is close to machine/ it consists of mnemonics and need assembler to translate/ it needs compiler or interpreter and it is close to human.

Example:

Machine Language	Assembly Language	High Level Language
Machine language is also called binary language. It is the language of 0's and 1's and does not need to be translated.	Assembly language is also called low level language. It is the language made up of ~~0's 1's~~ symbols. It needs an assembler to convert in into machine language.	High level language is closer to human language. It uses digits, symbols, characters and alphabets. It needs a compiler or an interpreter to translate.

Weaker responses showed that candidates did not know the main characteristic of each type of programming language that makes it distinct from others and wrote general or irrelevant characteristics such as easy to modify/ easy to learn/ easy error detection etc.

Example:

Machine Language	Assembly Language	High Level Language
Need translator Difficult to use	Easy to modify Easy to learn.	Easy error detection.

Question 5b:

What is the purpose of using **void** before **main( )** and **void** inside **brackets** in the given statement?

void main(void)

Using 'void' Before main( )	Using 'void' Inside Brackets

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

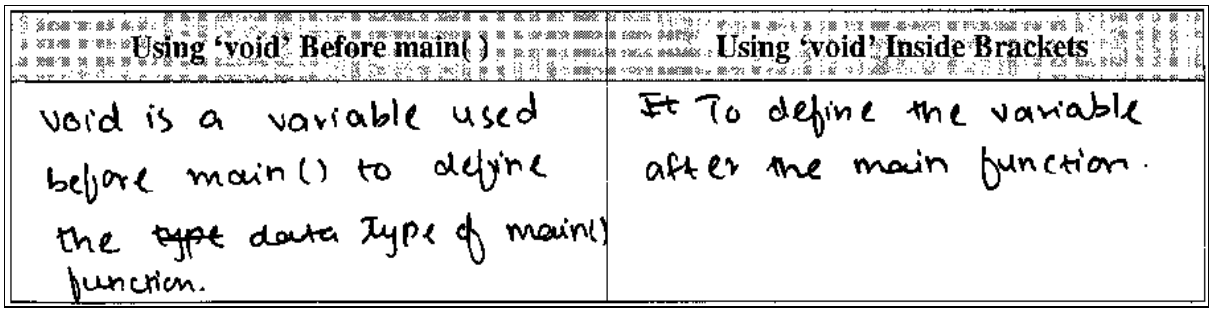
Better responses exhibited good understanding about return value by a function and passing arguments to a function and managed to write the purpose of using void before the main function and within main function such as the use of void before main function indicates that the main function will not return any value and the use of void inside the brackets of the main function means that the function does not have any arguments.

Example:

Using 'void' Before main( )	Using 'void' Inside Brackets
This void means that this main() function will not return anything or has no return type.	This void means this main() function would not take any sort of parameters in it. or in other words no parameter can be passed in this function

Weaker responses showed that candidates did not know about return value of a function and passing arguments to a function due to which they were unable to write the correct purpose of using void before main function and void within brackets of the main function such as void before the main function tells the data type of function and void within the function defines variable. A few of these responses depicted confusion between return value and passing argument and swapped both purposes.

Example:

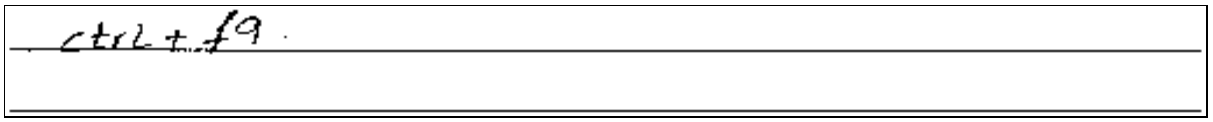


Question 6a:

What is the keyboard shortcut to run a C program in Turbo C integrated development environment (IDE)?

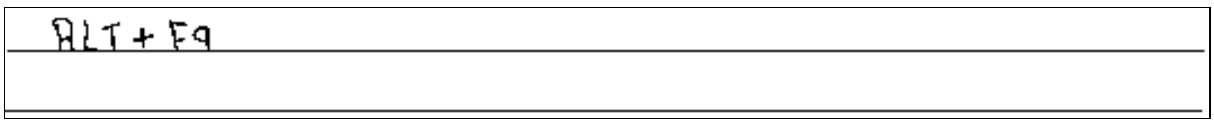
Better responses showed familiarity with Turbo C integrated development environment and wrote the correct keyboard shortcut to run a C program, i.e. Ctrl + F9.

Example:



Weaker responses depicted that candidates were unfamiliar with Turbo C integrated development environment and wrote incorrect or irrelevant keyboard shortcut such as Alt + F9 which is a shortcut to compile a C program/ Control + F9 instead of Ctrl + F9/ Alt + Ctrl + F9 etc.

Example:



Question 6b:

Write a C program that would input values of radius (r) and height (h), calculate and output the surface area of the right circular cone.

The formula to calculate the surface area of the right circular cone is:

$$\text{Area} = \pi r\left(r + \sqrt{h^2 + r^2}\right) \text{ where } \pi = 3.142$$

Better responses demonstrated strong concepts of the basics of C programming and were able to declare variables, input and output values in their program. Moreover, they were able to do the main task of this question, i.e. converting the given mathematical formula into C language code.

Example:

```
#include<stdio.h>    //including libraries
#include<conio.h>
#include<math.h>
main(){
float r,h,x, area ;
printf(" Enter radius and height");
scanf(" %f %f ",&r, &h);
X= h*h + r*r;    //calculating area
area = 3.142*r*(r+ sqrt(x));
printf(" Surface area of
        right circular
        cone is %.2f ", &area);
getch();
}
```

Weaker responses mostly managed to declare variables and/ or display output but they failed to write correct coding for taking input and converting the given mathematical formula into C language code.

Example:

```
#include(stdio.h)
#include(math.h)
        int main()
        {float r,h,π=3.142;
        Printf("Enter radius and height:");
        scanf("%f",&r,&h);
            { Root1= sqr(h) + sqr(r) }
        Printf("π*r(r+Root1)");
    }
```

Question 7:

a. Describe the purpose of using a loop in a program.

b. Write the general syntax of **While** loop and **If** statement.

c. What would be the output of the given code if the input is the first five integers, i.e. 1, 2, 3, 4, 5?

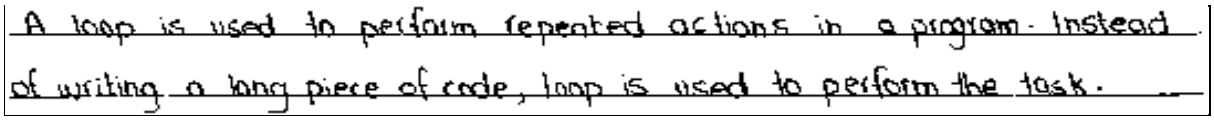
```
#include<stdio.h>
#include<conio.h>
void main( )
{
int n = 5;
float a, i, m, x = 0;
clrscr( );

for(i = 1; i <= n; i++)
{
 scanf("%f", &a);
 x = x + a;
}
m = x / n;
printf("\n%f", m);
getch( );
}
```

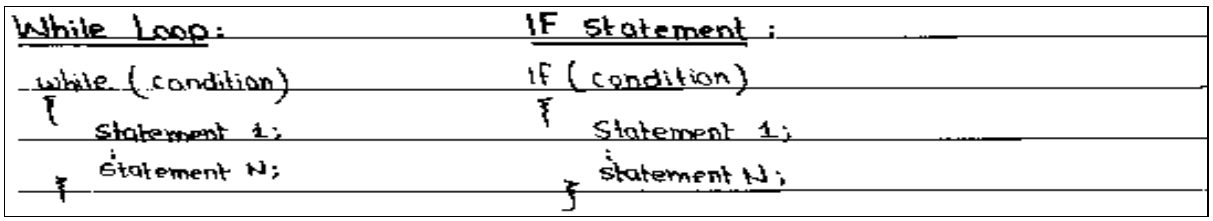
d. Infer the purpose of the given program.

Better responses demonstrated a good understanding of loops and selection statements and applied their understanding to complete the application level tasks given in this question, i.e. determining the output and inferring the purpose of the program. For example, loops are used to repeat a code segment/ the output of the program is 3/ the purpose of the program is to calculate the average of 5 numbers.

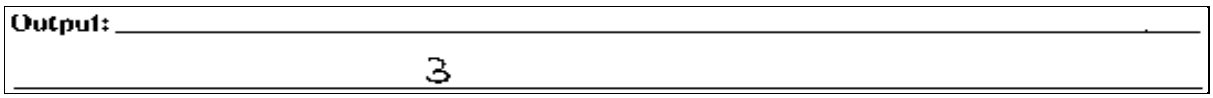
Example (a):


A loop is used to perform repeated actions in a program. Instead of writing a long piece of code, loop is used to perform the task.

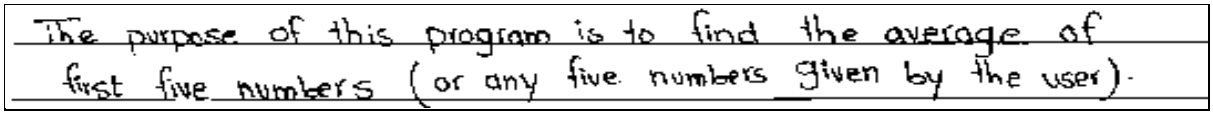
Example (b):



Example (c):

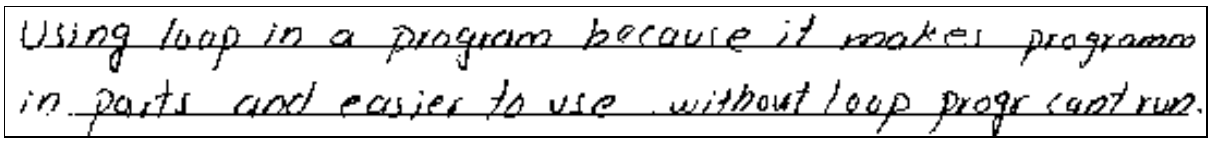


Example (d):

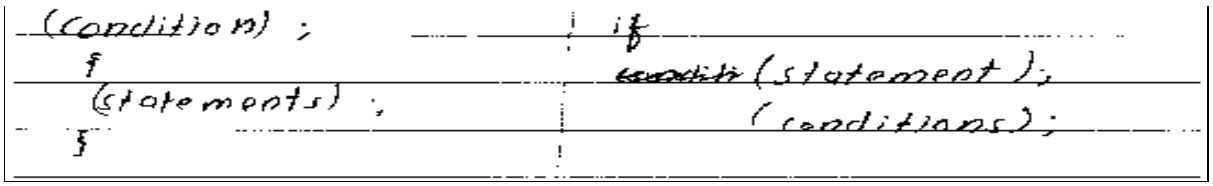


Weaker responses showed lack of understanding about the loops and selection statements due to which they did not determine the output and infer the purpose of program correctly and wrote such as a loop divides a program into parts/ output is 6, 7, 8, 9, 10 or 2/ the purpose of the program is to count numbers in ascending order. A few of these responses managed to describe the purpose of using loops in a program.

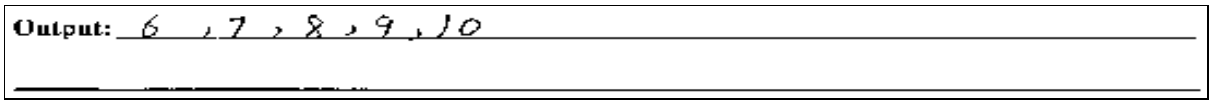
Example (a):



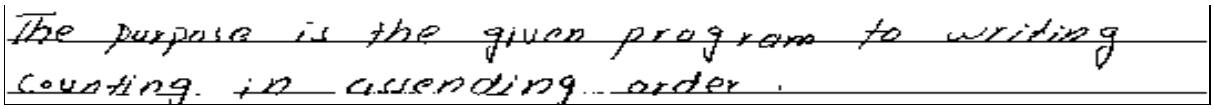
Example (b):



Example (c):



Example (d):



Question 8:

Complete the given code for a user-defined function named **Square** which would calculate the square of any input number and invoke (call) it in the main function.
The formula to calculate the square of a number **n** is given below.
Square = n × n

```
#include<stdio.h>
//Declare square function below
[                                                    ]
int main( )
{
   float m, n;
   printf ( "\nEnter some number for finding square \n");
   scanf ( "%f", &m );
   // Call function below
    [                                                    ]
 printf ( "\nSquare of the given number %f is %f",m,n );
}
 //Define function below
[                                                    ]
[                                                    ]
[                                                    ]
[                                                    ]
[                                                    ]
```

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

Better responses displayed candidates' ability to apply the concepts of user-defined functions in C programming in the scenario given in this question and declared, defined and invoked the function by writing correct C language code.

Example:

```
#include<stdio.h>
//Declare square function below
int Square (int num);
int main( )
{
   float m, n;
   printf ( "\nEnter some number for finding square \n");
   scanf ( "%f", &m );
   // Call function below
   n = Square (m);
   printf ( "\nSquare of the given number %f is %f",m,n
}
//Define function below
int Square (int num)
{   int ans;
      ans = num * num;
    return (ans);
}
```

Weaker responses failed to write the code to declare, invoke (call) and create the user-defined function in C programming in the scenario given in this question. These responses did not use the variables given in the question to invoke the function. A few of these responses wrote description instead of coding.

Example:

```
#include<stdio.h>
//Declare square function below
float main(Square)
int main( )
{
   float m, n;
   printf ( "\nEnter some number for finding square \n");
   scanf ( "%f", &m );
   // Call function below
   Square=n*n
   printf ( "\nSquare of the given number %f is %f",m,n );
}
//Define function below
```

For example if the user enters 3 then the output will be

```
Enter some number for finding square
3
Square of the given number 3 is 9.
```

Question 9:

Read the given C language code.

```
#include <stdio.h>
#include <conio.h>
{
        FILE *fp;
        char ch;
        fp = fopen("MYFILE.TXT", "w");
        while((ch = getche( )) != '\r')
                putc(ch,fp);
        fclose(fp);
}
```

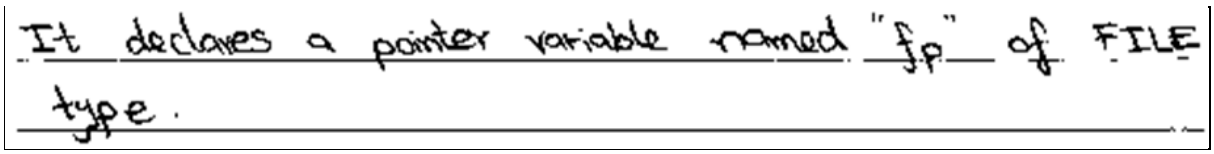
Interpret the purpose of the following code segments from this code.

i. FILE *fp;
ii. fp = fopen("MYFILE.TXT", "w");
iii. while((ch = getche( )) != '\r')
iv. putc(ch,fp);

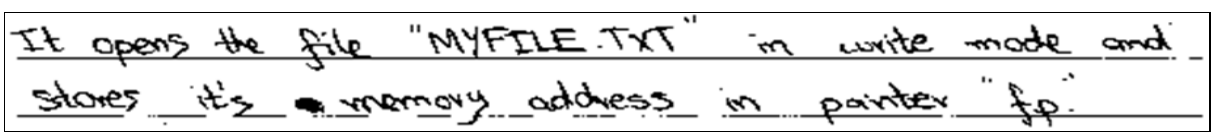
Better responses exhibited a good understanding of the file management and its built-in functions in C language and applied their understanding to interpret the purpose of different code segments given in the question. For example, FILE *fp defines the file structure fp. The file structure is declared in the header file stdio.h/ fp = fopen("MYFILE.TXT", "w") tells operating system to open a file called MYFILE.TXT in write mode/ while((ch = getche( )) != '\r') command tells that While loop condition will become false when carriage return (\r) is pressed and loop will stop/ putc(ch,fp); writes the character stored in ch to a file pointed by file pointer fp.

It is to be noted that most of these responses wrote that the loop will stop when \r is entered instead of when the enter button is pressed in part (iii).

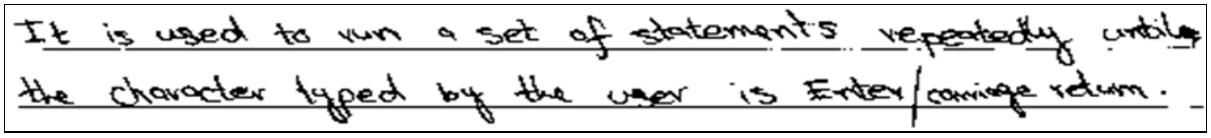
Example (i):



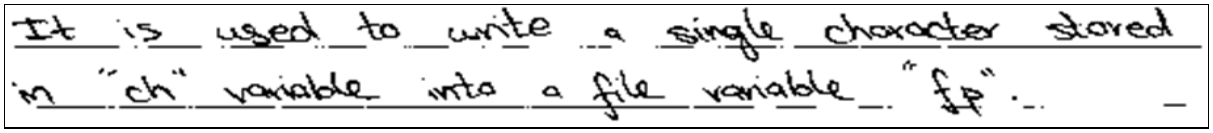
Example (ii):



Example (iii):

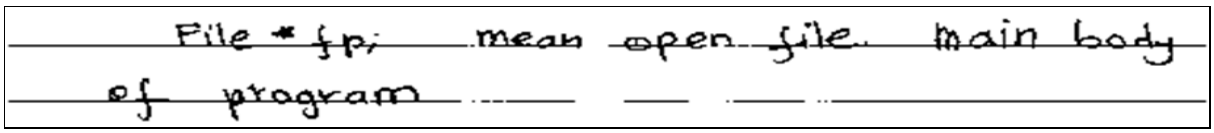


Example (iv):

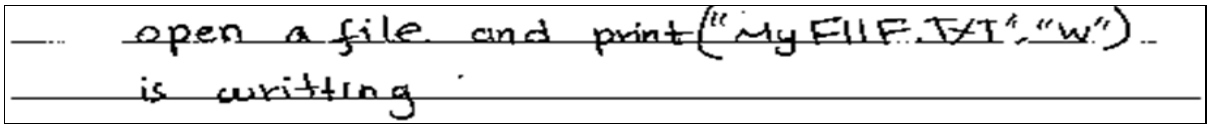


Weaker responses showed poor conceptual understanding of the file management and its built-in functions in C language and wrote incorrect purpose of code segments such as File *fp opens a file/ fp = fopen("MYFILE.TXT", "w") opens a file and print/ in While((ch = getche( )) != '\r') command the While loop will continue for any character input/ putc(ch,fp) reads data from user.

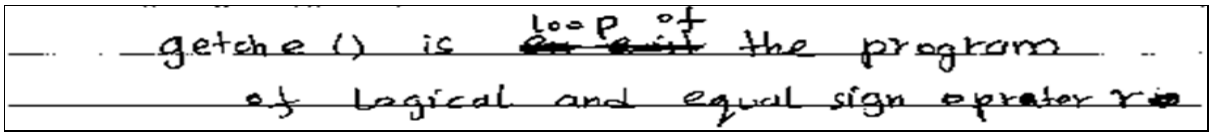
Example (i):



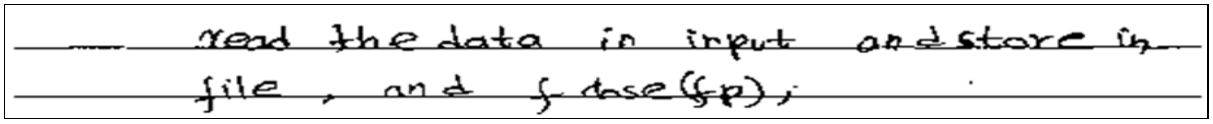
Example (ii):



Example (iii):



Example (iv):



Extended Response Questions (ERQs)

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

Question 10a:

Write a C program for calculating the roots of a quadratic equation using the quadratic formula. Also, determine whether the roots are real or imaginary. Apply suitable conditions and print an appropriate message.

The quadratic formula to calculate the roots of a quadratic equation is given below.

Roots = $\frac{-b \pm \sqrt{d}}{2a}$, where $d = b^2 - 4ac$; Roots are real when $d \geq 0$ otherwise roots are imaginary.

(**Note:** Most of the candidates opted part **b** of this question. However, they performed almost equally well in both parts.)

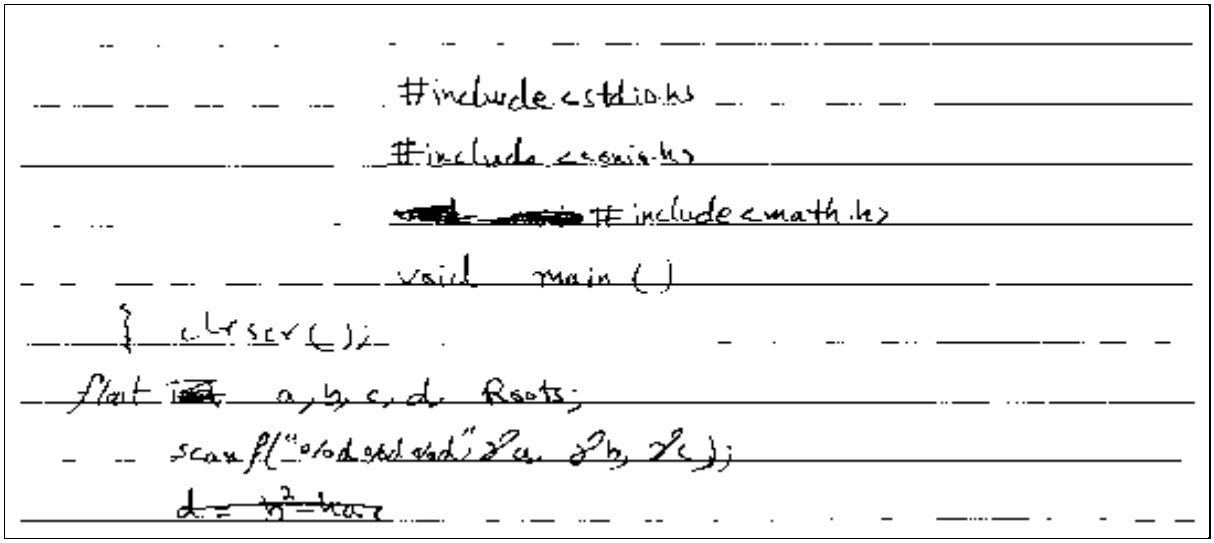
Better responses understood the given problem well and built the logic to solve this problem by converting the given mathematical equation into C language statement. Moreover, these responses wrote the correct code to determine whether the roots are real or imaginary based on the criteria given in the question.

Example:

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
main()
{
 int a,b,c,d;
  float x1,x2;
 printf("enter values of a,b and c from quadratic equation:");
  scanf("%d %d %d",&a,&b,&c);
d = b*b - 4*a*c;
x1 = -b + Sqrt(d) /2*a ;
x2 = -b - Sqrt(d) /2*a;
printf("roots of quadratic equation are=%f %f",x1,x2);
if(d >= 0)
printf(" roots are real");
else
 printf(" roots are imaginary");
 getch();
}
```

Weaker responses mostly understood the question but due to lack of practice and weaker concepts of programming, they were not able to build the logic required to solve the given problem. Such responses used mathematical operators in C language statement and use incorrect condition with selection statement to decide whether the roots are real or imaginary.

Example:



```
#include<stdio.h>
#include<conio.h>
#include<math.h>
void main()
{clrscr();
float a,b,c,d, Roots;
scanf("%d%d%d",&a, &b, &c);
d= b*b-4*a*c;
if (d>0)
printf("\nThe roots are Real");
Else
Printf("\nThe roots are imaginary");
Roots = (-b+-sqrt(d))/2*a;
Printf("\n the answer = %d", Roots);
Getch();
}
```

Question 10b:

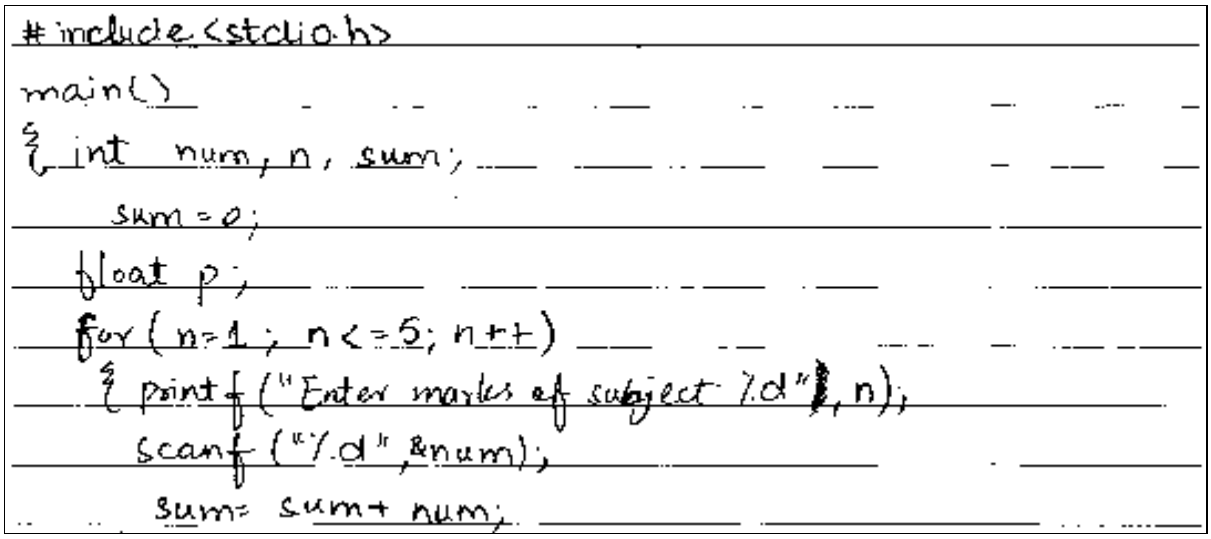
Write a C program to accept the marks of a student for FIVE subjects, calculate and print the overall percentage and print the student's grade according to the criteria in the given table.

(**Note:** Total marks of subjects are 500.)

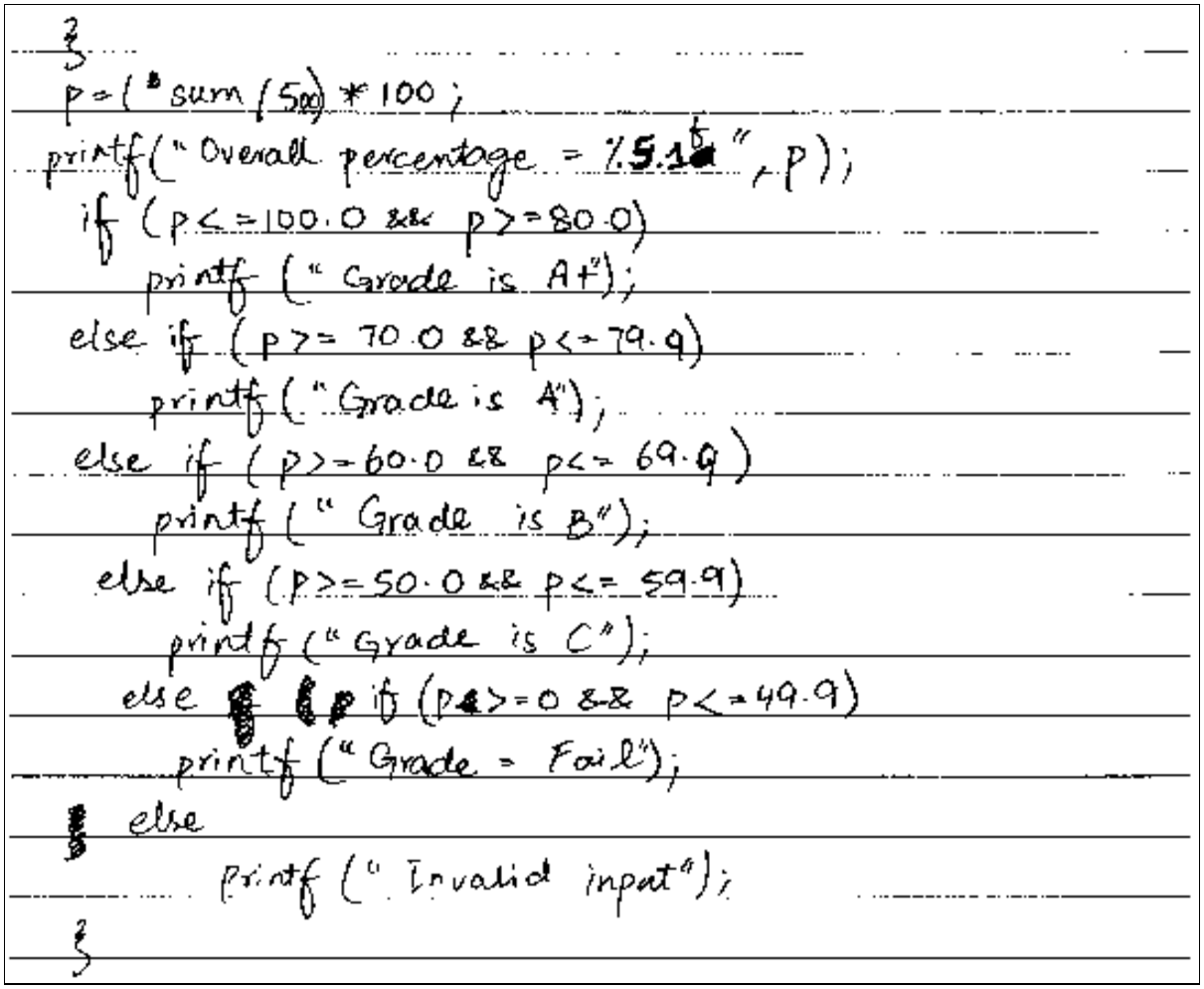
Percentage	Grade
80.0 to 100.0	A+
70.0 to 79.9	A
60.0 to 69.9	B
50.0 to 59.9	C
0 to 49.9	Fail

Better responses declared variables with appropriate data types to store marks of each subject and input marks of 5 subjects and stored them in declared variables. Furthermore, these responses wrote the C code to calculate the percentage and check percentage range using IF…ELSE statement and logical AND operator and printed the grades according to the criteria given in the question.

Example:

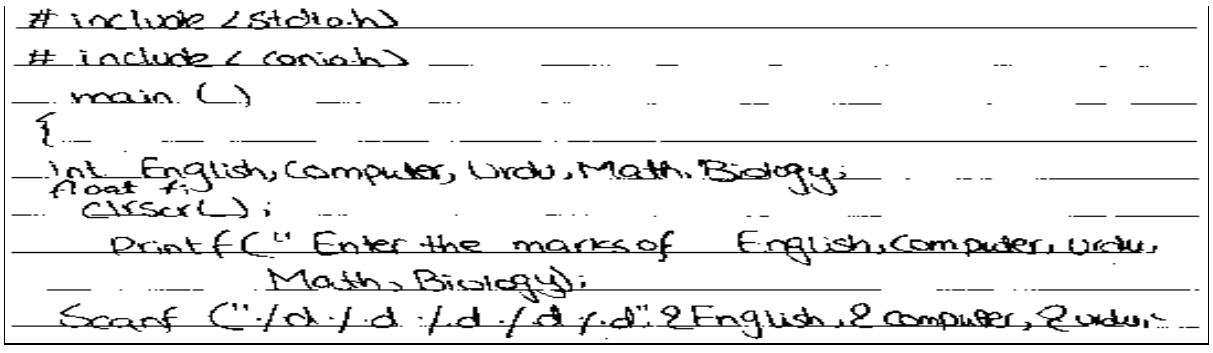


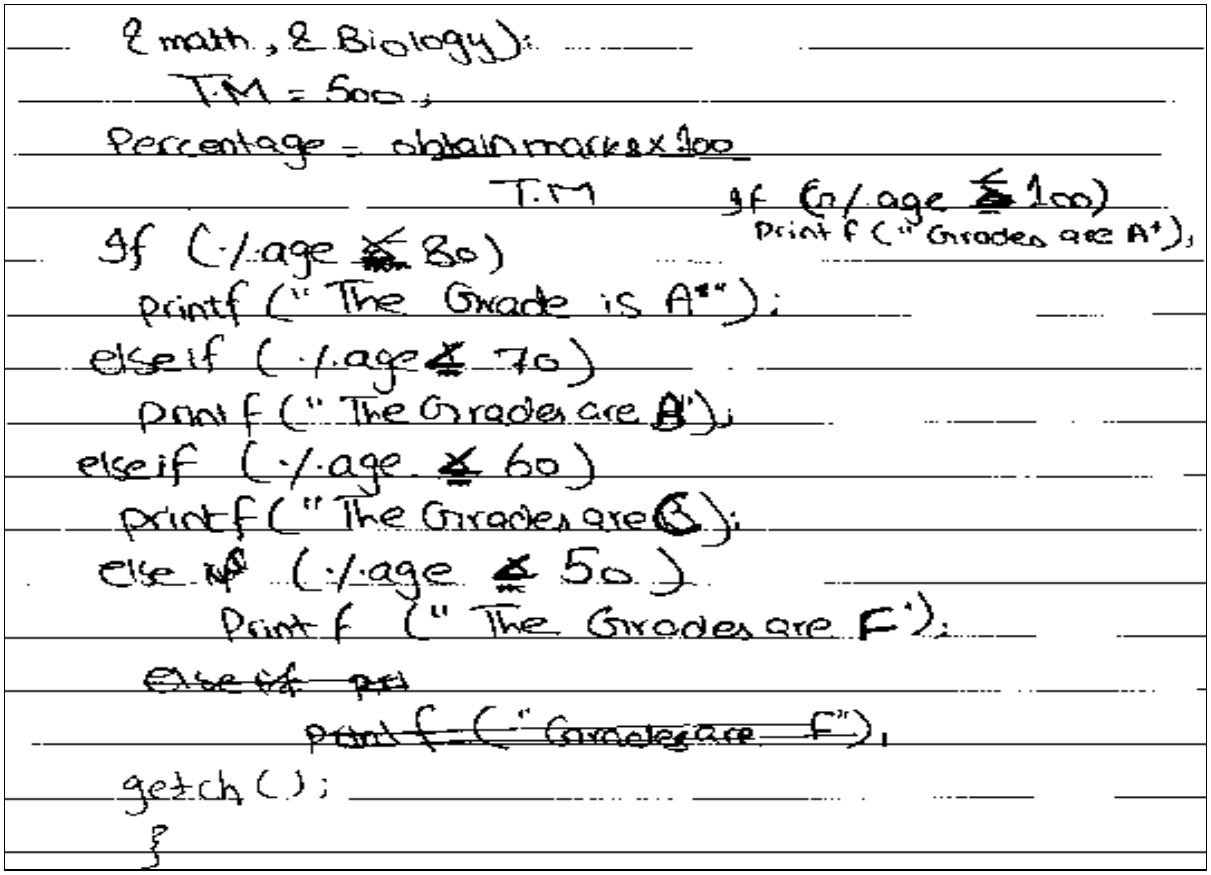
```
#include<stdio.h>
main()
{ int num, n, sum;
   sum=0;
  float p;
  for(n=1; n<=5; n++)
   { printf("Enter marks of subject %d", n);
     scanf("%d", &num);
     sum= sum+ num;
```



Weaker responses did not understand the question due which they were not able to build the logic required to solve the given problem. Such responses were not able to write the code to calculate the percentage and check percentage range using the correct syntax of IF…ELSE statement. These responses did not use logical AND operator in the IF…ELSE condition. However, a few of these responses managed to declare variables with appropriate data types and input values from the user.

Example:





Note: Less than 1% of the candidates solved questions 11-16 that were based on Visual Basic (VB) programming. It was good to see that candidates tried to attempt VB Programming Section. However, candidates obtained very low 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 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.