Province of the
EASTERN CAPE
EDUCATION

NATIONAL SENIOR CERTIFICATE

GRADE 11

NOVEMBER 2022

INFORMATION TECHNOLOGY P1 MARKING GUIDELINE (EXEMPLAR)

MARKS: **150**

This marking guideline consists of 12 pages.

NAME OF LEARNER:				
TOTAL QUESTION 1	**TOTAL QUESTION 2**	**TOTAL QUESTION 3**	**TOTAL QUESTION 4**	**TOTAL**
/40	**/40**	**/40**	**/30**	**/150**

QUESTION 1		MAX MARKS	MARKS ACHIEVED
1.1	**Menu option New Registration** pnlRegister.visible := true ✓	**1**	
1.2	**Button [btnQ1_2] Register** Get date ✓ Extract the year ✓ add one ✓ Compile expiry date✓ Get access string from cmbaccess ✓ Get input from four spinedits ✓as integers ✓ Calculate total ✓using four correct rand values ✓ If cmbaccess.itemindex = 1 (access is Regional) ✓ Then subtract 40% from total due ✓ Use a showmessage component ✓ With correct wording, use #13 ✓ use access and expiry date ✓	**14**	
1.3	**Menu option Log In** Enable pnllogin ✓ Clear the two edit boxes ✓ Set focus to edtname ✓ pnllogin.font.color = clgreen ✓ pnllogin.color = clcream ✓	**5**	

1.4	**Button [btnQ1_4] Log In** Set constant alphabet string ✓ Get the name ✓ Get password converted to uppercase (or alphabet string includes lowercase letters) ✓ Initialise new string ✓ Loop from 1 to length of password ✓ Use if statements or case ✓ Replace space with * ✓ Replace Z with A ✓ Replace A to Y with correct character from alphabet ✓ Add three to character position ✓ Join new character to new string ✓ Join all other characters ✓ (not A to Z and not space) to new string ✓ Join last three characters of name ✓to end of new string ✓ display in edtencrypt ✓	**16**	
1.5	**Menu option Log Out** If - then statement ✓ Message box ✓ Input = false ✓ Application.terminate ✓	**4**	
	Question 1 Total	**40**	

QUESTION 2		MAX MARKS	MARKS ACHIEVED
2.1	**Button [Q2.1 Read and Display]** <u>Read from text file</u> set icountarr to 0 ✓ assignfile to text file variable ✓ reset ✓ loop until end of text file ✓ readln statement ✓ increment icountarr ✓ get tree name ✓ delete string to get quantity converted to integer ✓ and store in arrqty using array counter as index ✓ if tree name contains quotation marks ✓ then delete first character of tree name ✓ and delete last character tree name ✓ store tree name in arrtrees using array counter as index ✓ <u>Sorting two arrays</u> outer loop from 1 to array counter - 1 ✓ Inner loop from outer loop counter + 1 to array counter ✓ If arrqty[outer] > arrqty [inner] then ✓ Set integer temp variable = arrqty [outer] ✓ set arrqty [outer] = arrqty [inner] ✓ set arrqty [inner] = integer temp variable ✓ do the same swap algorithm for arrtrees using string temp variable ✓ <u>calculate total</u> initialise total variable ✓ loop from 1 to icountarr ✓ set total = total + arrqty using loop index ✓ <u>Display arrays</u> Loop from 1 to icountarr ✓ use richedit ✓ display contents of arrays using loop counter as index arrtrees, ✓arrqty converted to string ✓ using tab stop ✓ <u>Display total</u> display total in richedit with correct message ✓ total converted to string ✓ display array counter converted to string ✓ in edtprotected ✓	**32**	
2.2	**Button [Q2.2 Endangered list]** Assignfile with correct file name ✓ Rewrite statement ✓ Loop from 1 to array counter ✓ If arrqty contents using loop counter as index < 100 ✓ Then get string from arrtrees ✓ Display string in richedit ✓ Write string to the text file ✓ Closefile statement after loop ✓	**8**	
	Question 2 Total	**40**	

QUESTION 3		MAX MARKS	MARKS ACHIEVED
3.1	**Button [Q3.1 Western Cape]** Go to first record ✓ Loop to end of the table ✓ If Area = 'Western Cape' ✓ Then display ScientificName ✓ in richedit ✓ Go to next record before end of loop ✓	**6**	
3.2	**Button [Q3.2 Iconic and Protected]** Go to first record ✓ Loop to end of the table ✓ If Iconic = true ✓ and ✓ Description = 'Protected' ✓ Then display OtherName✓ in richedit ✓ Go to next record before end of loop ✓	**8**	
3.3	**Button [Q3.3 Delete Invasive Trees]** Go to first record ✓ Loop to end of the table ✓ if Description ✓= 'invasive' then ✓ Delete ✓ else ✓ Go to next record before end of loop ✓	**7**	
3.4	**Button [Q3.4 Red Favourites]** Go to first record ✓ Loop to end of the table ✓ Get Othername and store in string variable ✓ If first 3 characters ✓ = 'Red ✓ Then edit mode✓ Set Favourite = true ✓ post ✓ Go to next record before end of loop ✓	**9**	
3.5	**Button [Q3.5 Indigenous Average]** Set counter to 0 and total to 0 ✓ Go to first record and Loop to end of the table ✓ If Indigenous= true ✓ and Counted <> Null ✓ Then Add 1 to counter ✓ Add Counted field to total variable ✓ Go to next record before end of loop ✓ Calculate average: total / counter ✓ Display in richedit ✓ Correct description, Rounded average and converted to string ✓	**10**	
	Question 3 Total	**40**	

QUESTION 4		MAX MARKS	MARKS ACHIEVED
4.1	**Button [Question 4.1]** Use randomrange(1000,10000) or random(10001) + 1000 to generate a random number ✓ Display correct message and random number converted to string ✓ in memo box ✓ Outer Loop from 1 to the random number ✓ Initialize total variable ✓ Inner Loop from to 1 to outer loop counter ✓ If innerloop counter is a factor ✓of outerloop counter ✓ Then add inner loop counter to total ✓ Subtract outer loop counter from total ✓ If total = outer loop counter ✓ Then display outer loop counter converted to string in memo box ✓	**12**	
4.2	**Button [Question 4.2]** Use Val to test if edit box contains integer ✓ If not then show message ✓ and exit ✓ Get number from edit box converted to integer ✓ Set a string variable to an empty string ✓ Loop until number = 0 ✓ Set remainder variable = remainder after number is divided by 16 (use MOD) ✓ use if statement or Case ✓ set string for hexadecimal number to correct letters ✓ A to F ✓ for each number in range 10 to 15 ✓ Add letter to front of string variable ✓ Else ✓ if remainder < 10 ✓ Then Add remainder variable converted to string ✓ to the front of the string variable ✓ use DIV to divide number by 16 ✓ display the string variable in the edit box edtHexadecimal ✓	**18**	
	Question 4 Total	**30**	

SAMPLE SOLUTIONS

QUESTION 1

```
//Question 1.1
procedure TQuestion1.Ques1_1Click(Sender: TObject);
begin
//Add your code below:
   pnlregister.visible := true;

end;

 //Question 1.2
procedure TQuestion1.Ques1_2Click(Sender: TObject);
begin
 //Add your code below:
 pnllogin.enabled := true;
 edtname.Clear;
 edtpassword.Clear;
 edtname.SetFocus;
 pnllogin.font.color := clgreen;
 pnllogin.Color := clcream;
end;

//Question 1.3
procedure TQuestion1.Ques1_3Click(Sender: TObject);
var sans : string;
begin
 //Add your code below:
//if messagedlg('Are you sure you want to log out?',mtwarning,[mbYes, mbNo],0) = mryes
then
      //application.Terminate;

//Alternative answer
 sans := inputbox('Are you sure you want to log out?', 'Y or N','');
 if uppercase(sans) = 'Y' then
     application.Terminate;
end;

//Question 1.2
procedure TQuestion1.btnQ1_2Click(Sender: TObject);
var sdate, sexpire : string;
saccess : string;
iadult, isen, istud, ischolar : integer;
rtotal : real;
begin
//Add your code below:
 sdate := edtdate.Text;
 sexpire := inttostr(strtoint(copy(sdate,1,4))+1)+copy(sdate,5,6);
 saccess := cmbaccess.Items[cmbaccess.ItemIndex];
 iadult := sedadult.Value;
 isen := sedsenior.Value;
```



```
 istud := sedstudent.Value;
 ischolar := sedscholar.Value;
 rtotal := (iadult * 500) + (isen*400) + (istud *300) + (ischolar *80);
 if cmbaccess.itemindex = 1 then
      rtotal := rtotal - rtotal *40/100 ;

showmessage('You owe ' + floattostrf(rtotal,ffcurrency,8,2)
 + ' for ' + saccess +  ' access.' + #13 + ' Expiry date = ' + sexpire);

Ques1_3.Click;

end;

//Question 1.4
procedure TQuestion1.btnQ1_4Click(Sender: TObject);
var sname, spass, snew , sword: string;
k, ipos : integer;
const
salpha =  'ABCDEFGHIJKLMNOPQRSTUVWXYZ';
begin
//Add your code below:
  sname := edtname.Text;
  spass := uppercase(edtpassword.Text);
  snew := '';
  for k := 1 to length(spass) do
    begin
    case spass[k] of
      ' ' : snew := snew + '*';
      'Z' : snew := snew + 'A';
      'A'..'Y' :
      begin
        ipos := pos(spass[k], salpha);
        ipos := ipos + 3;
        snew := snew + salpha[ipos]
      end
      else
      snew := snew + spass[k];
    end;
  end;
edtencrypt.Text := snew+ copy(sname,length(sname) - 2,3);
end;
```

QUESTION 2

```
///Question 2.1   32 marks
procedure TfrmQuestion2.btnQ2_1Click(Sender: TObject);
var myfile : textfile;
sline, stree,sqty, stemp : string;
iqty, itotal,k,l, itemp : integer;
begin
redout.Clear;
if not fileexists('Trees.txt') then
```

```
begin
  showmessage('file not found');
  exit;
end;
assignfile(myfile,'Trees.txt');
reset(myfile);
icountarr := 0;
while not eof(myfile) do
   begin
   readln(myfile,sline);
   stree := copy(sline,1, pos('#',sline)- 1);
   delete(sline,1, pos('#',sline));
   sqty := sline;

   inc(icountarr);
   arrqty[icountarr] := strtoint(sqty);
    if stree[1] = "" then
        begin
        delete(stree,1,1);
        delete(stree,length(stree),1);
        end;
    arrtrees[icountarr]:= stree;
  end;
for k := 1 to icountarr - 1 do
     for l := k + 1 to icountarr do
        begin
          if arrqty[k] > arrqty[l] then
             begin
               itemp := arrqty[k];
               arrqty[k] := arrqty[l];
               arrqty[l] := itemp;
               stemp := arrtrees[k];
               arrtrees[k] := arrtrees[l];
               arrtrees[l] := stemp;
             end;
        end;
itotal := 0;
for k  := 1 to icountarr do
 itotal := itotal + arrqty[k];
for k  := 1 to icountarr do
 redout.Lines.Add(arrtrees[k]  + #9 + inttostr(arrqty[k]));
redout.Lines.Add('');
redout.Lines.Add('Total number of protected trees: ' + inttostr(itotal));
  edtprotected.Text := inttostr(icountarr);
end;

 ///Question 2.1   8 marks
procedure TfrmQuestion2.btnQ2_2Click(Sender: TObject);
var k : integer;
sline : string;
tfile : textfile;
begin
```



```
redout.Clear;
assignfile(tfile,'Endangered.txt');
rewrite(tfile);
 for k := 1 to icountarr do
    begin
      if arrqty[k] < 100 then
         begin
           sline := arrtrees[k];
           writeln(tfile,sline);
           redout.Lines.Add(sline);
         end;
    end;
 closefile(tfile);
end;
```

QUESTION 3

```
////Question 3.1  6 Marks
procedure TfrmQuestion3.btnQ3_1Click(Sender: TObject);
begin
reddisplay.Clear;
tbltrees.First;
while not tbltrees.eof do
   begin
     if tbltrees['Area'] = 'Western Cape' then
        reddisplay.lines.add(tbltrees['ScientificName']);
   tbltrees.Next;
   end;
end;

////Question 3.2  8 Marks
procedure TfrmQuestion3.btnQ3_2Click(Sender: TObject);
begin
 reddisplay.Clear;
tbltrees.First;
while not tbltrees.eof do
   begin
     if (tbltrees['Iconic'] = true) and (tbltrees['Description'] = 'Protected') then
        reddisplay.lines.add(tbltrees['OtherName']);
   tbltrees.Next;
   end;
end;

////Question 3.3  7 Marks
procedure TfrmQuestion3.btnQ3_3Click(Sender: TObject);
begin
tbltrees.First;
while not tbltrees.eof do
   begin
     if tbltrees['Description'] = 'invasive' then
        tbltrees.Delete
      else
```



```
   tbltrees.Next;
   end;
end;

////Question 3.4  9 Marks
procedure TfrmQuestion3.btnQ3_4Click(Sender: TObject);
 var sn : string;
begin
  tbltrees.First;
while not tbltrees.eof do
   begin
     sn := tbltrees['OtherName'];
     if copy(sn,1,3) = 'Red' then
     begin
        tbltrees.edit;

        tbltrees['Favourite'] := true;
        tbltrees.Post;
     end;
   tbltrees.Next;
   end;
end;

////Question 3.5  10 Marks
procedure TfrmQuestion3.btnQ3_5Click(Sender: TObject);
var itotal, inum : integer;
rave : real;
begin
reddisplay.Clear;
  tbltrees.First;
  itotal := 0;
  inum := 0;
while not tbltrees.eof do
   begin
    if (tbltrees['Indigenous'] = true) and (tbltrees['Counted'] <> null) then
     begin
       itotal := itotal +  tbltrees['Counted'];
       inc(inum);
     end;
   tbltrees.Next;
   end;
 rave := itotal/inum;
 reddisplay.Lines.Add('Average Indigenous trees found:' + inttostr(round(rave)));
end;
```

QUESTION 4

```
/// Question 4.1   12 marks
procedure TfrmQuestion4.btnQ4_1Click(Sender: TObject);
var
iperfect, isum, m, iran : integer;
begin
```



```
//Enter your code here:
iran := randomrange(1000,10000);
memdisplay.Lines.Add('Perfect Numbers in the range from 1 to ' + inttostr(iran));
  for iperfect := 1 to iran do
      begin
      isum := 0;
      for m := 1 to iperfect  do
        begin
         if iperfect mod m = 0 then
          isum := isum + m;
        end;
      isum := isum - iperfect;
      if isum = iperfect then
      memdisplay.Lines.Add(inttostr(iperfect))
      end;
end;

// Question 4.2  18 marks
procedure TfrmQuestion4.btnQ4_2Click(Sender: TObject);
var inum, icode, k, ihex : integer;
shex : string;
begin
val(edtdecimal.Text,inum, icode);
 if icode <> 0 then
     begin
       showmessage('Please enter a valid integer');
       exit;
     end;
 inum := strtoint(edtdecimal.Text);
  shex := '';
 while inum <> 0 do
    begin
      ihex := inum mod 16;
      case ihex of
        10 : shex := 'A' + shex;
        11 : shex := 'B' + shex;
        12 : shex := 'C' + shex;
        13 : shex := 'D' + shex;
        14 : shex := 'E' + shex;
        15 : shex := 'F' + shex;
      else
        shex := inttostr(ihex) + shex;
       end;
     inum := inum div 16
    end;
   edthexadecimal.Text := shex;
end;
```