

GAUTENG DEPARTMENT OF EDUCATION
SENIOR CERTIFICATE EXAMINATION

OCTOBER / NOVEMBER 2005
OKTOBER / NOVEMBER 2005

COMPUTER STUDIES HG
(First Paper: Practical)

TIME: 3 hours

MARKS: 100

INSTRUCTIONS:

- All questions must be answered.
 - Each question has a Delphi as well as a Pascal section. Answer only the sections applicable to you.
 - No components may be deleted from or added to the given Delphi forms.
 - Poor programming techniques will be penalised.
 - Save your work at regular intervals.
 - Your full examination number must appear on every page that is handed in.
-
-

QUESTION 1

Delphi: (Question 1)

A data file called '**membersD.dat**' is stored on your examination disk. Each record within the file has been declared to consist of the following FOUR fields:

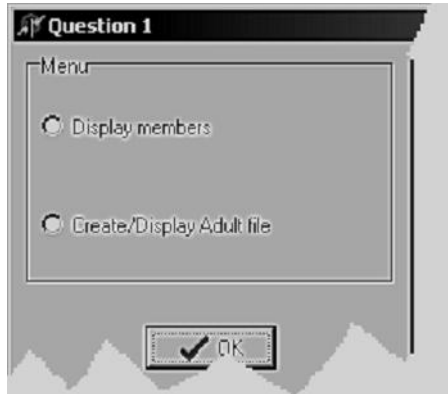
Initials à	string of 4 characters
Surname à	string of 30 characters
Adult à	Boolean value {True represents <i>adult</i> ; False represents <i>child</i> }
Balance à	Real value

The file mentioned above contains data on all members of a sports club. In future the club will only allow adult members. A new data file needs to be created that will only contain the data of the adult members.

Open the file '**Q1P.dpr**' in Delphi, go to File|Save As... and save the *unit* as '**Q1U_XXXX.pas**' (XXXX represents the last four digits of your examination number). Now go to File|Save Project As... and save the *project* as '**Q1P_XXXX.dpr**'.

- 1.1 Rename *Radiogroup1* to *rgpChoices* and change its *Caption* and *Items* properties to correspond with the figure below. Also add your examination number to the *Caption* of the form next to "Question 1".

(3)



When the user clicks on [OK] the program should do the following:

- 1.2 If *Display members* has been selected, the contents of `membersD.dat` must be read and displayed in *redOutput* as follows: (20)



- 1.3 If *Create/Display Adult file* has been selected, only the adult members that are contained within `membersD.dat` must be written to a new file called `adult.dat`. The records written to the new file must be written WITHOUT the Adult field. (Only the Initials, Surname and Balance fields must be written to `adult.dat`). Every member that is written to `adult.dat` must also be displayed in *redOutput* as follows: (15)



The following must be handed in for Question 1.1 to 1.3 (Delphi):

- A printout of **Q1U_XXXX.pas**.
- An "Alt | Print Scrn" of the first report (All members).
- An "Alt | Print Scrn" of the second report (Adult members).

- 1.4 Start a new application in Delphi and save the *unit* as **BalanceU_XXXX.pas** and the *project* as **BalanceP_XXXX.dpr** (**XXXX** represents the last four digits of your examination number). Design a form like the one below and add your examination number next to "Question 1.4" to the *Caption* of the form:



Expand Question 1.4 so that it will count the number of persons in '**membersD.dat**' who have a balance of R0.00 and also the number of persons who have a balance **greater than** R0.00 (Debit Balance).

- The name of the file that has to be read, must be entered into the *Edit*.
- After the user has entered the name of the file into the *Edit* and has clicked on **[OK]**, the output of the program must appear on the *Panels*, similar to the form given above.
- No extra subroutines must be written.

(12)

The following must be handed in for Question 1.4 (Delphi):

- A printout of **BalanceU_XXXX.pas**.
- An "Alt | Print Scrn" of the form displaying the output.

QUESTION 1: [50]

Pascal: (Question 1)

A data file called '**membersP.dat**' is stored on your examination disk. Each record within the file has been declared to consist of the following FOUR fields:

Initials à	string of 4 characters
Surname à	string of 30 characters
Adult à	Boolean value {True represents <i>adult</i> ; False represents <i>child</i> }
Balance à	Real value

The file mentioned above contains data on all members of a sports club. In future the club will only allow adult members. A new data file needs to be created that will only contain the data of the adult members.

Start with a new program in Turbo Pascal and save it as 'Q1_XXXX.pas' (XXXX represents the last four digits of your examination number). The program must adhere to the following:

1.1 The menu below must be displayed *repeatedly* until the user chooses option 3:

```
1. Display members.
2. Create/Display Adult file.
3. Quit.
```

(3)

1.2 When *Option 1* is chosen, the contents of `membersP.dat` must be read and displayed on screen in the following manner:

Initials	Surname	Adult	Balance
AB	DU PLESSIS	Yes	235.50
G	HOLTZHAUSEN	No	110.00
Q	SMITH	Yes	0.00
K	JONES	No	350.71
M	ZWANE	Yes	0.00
MW	DE KOKER	No	35.00
P	PETERS	Yes	766.67
A	MOLOI	No	25.00
J	VAN DER MERWE	Yes	0.00
JJ	LESSING	Yes	0.00
LCM	KRUGER	No	99.43
B	POTOMAC	Yes	0.00

(20)

1.3 When *Option 2* is chosen, only the adult members that are contained within `membersP.dat` must be written to a new file called `adult.dat`. The records written to the new file must be written WITHOUT the adult field. (Only the Initials, Surname and Balance fields must be written to `adult.dat`). Every member that is written to `adult.dat` must also be displayed on screen as follows:

Initials	Surname	Balance
AB	DU PLESSIS	235.50
Q	SMITH	0.00
M	ZWANE	0.00
P	PETERS	766.67
J	VAN DER MERWE	0.00
JJ	LESSING	0.00
B	POTOMAC	0.00

(15)

The following must be handed in for Question 1.1 to 1.3 (Pascal):

- A printout of `Q1_XXXX.pas`.

1.4 Start a new program in Turbo Pascal and save it as `XXXXblnc.pas` (XXXX represents the last four digits of your examination number).

Expand the program so that it will count the number of persons in 'membersP.dat' who have a balance of R0.00 and also the number of persons who have a balance **greater than** R0.00 (Debit Balance).

- The user must be prompted for the file name as soon as the program is run.
- No subprograms must be written.
- Output of the program must only be the following:

```
Zero Balance: 5
Debit Balance: 7
```

(12)

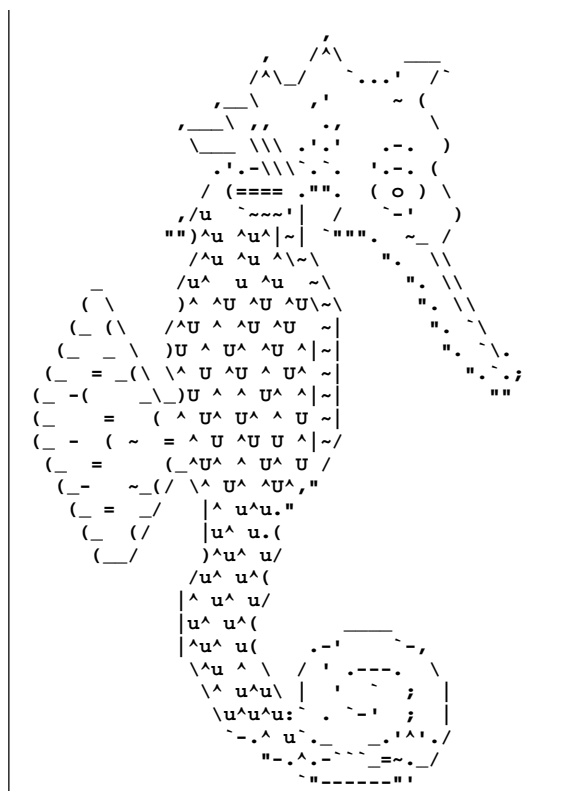
The following must be handed in for Question 1.4 (Pascal):

- A printout of `XXXXblnc.pas`.

QUESTION 1: [50]

QUESTION 2

ASCII-art is a kind of art where pictures are formed by using characters of text. An example of an ASCII-picture:



A text file containing hidden ASCII-pictures is stored on your examination disk. All the lines that belong to a certain picture will contain a specific code somewhere within that line. The first character of every line is a letter of the alphabet, which is an indication of the position of the line within the picture. The file is stored as 'mainfile.txt'.

Example:

The code ;;;; was used to hide the picture below.

All the lines in the text file that contain the code ;;;;

```

D      /      ;;;;      \
C      /      \ ;;;;
E      ;;;;  -----
a      :      : ;;;;
F      :      ;;;;:
A      / ;;;; \
G ;;;; :      :
B ;;;; /      \
b      ----- ;;;; ----
H      : ;;;; :
    
```

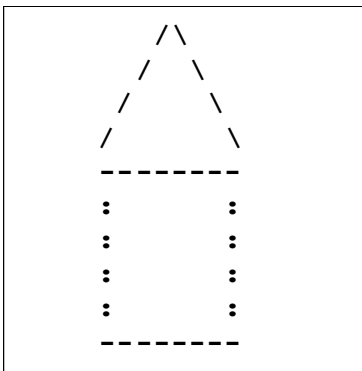
The lines above sorted alphabetically:

```

A      / ;;;; \
B ;;;; /      \
C      /      \ ;;;;
D      /      ;;;;      \
E      ;;;;  -----
F      :      ;;;;:
G ;;;; :      :
H      : ;;;; :
a      :      : ;;;;
b      ----- ;;;; ----
    
```

(Remember: 'A' < 'a')

The final picture after the code as well as the first character have been removed:



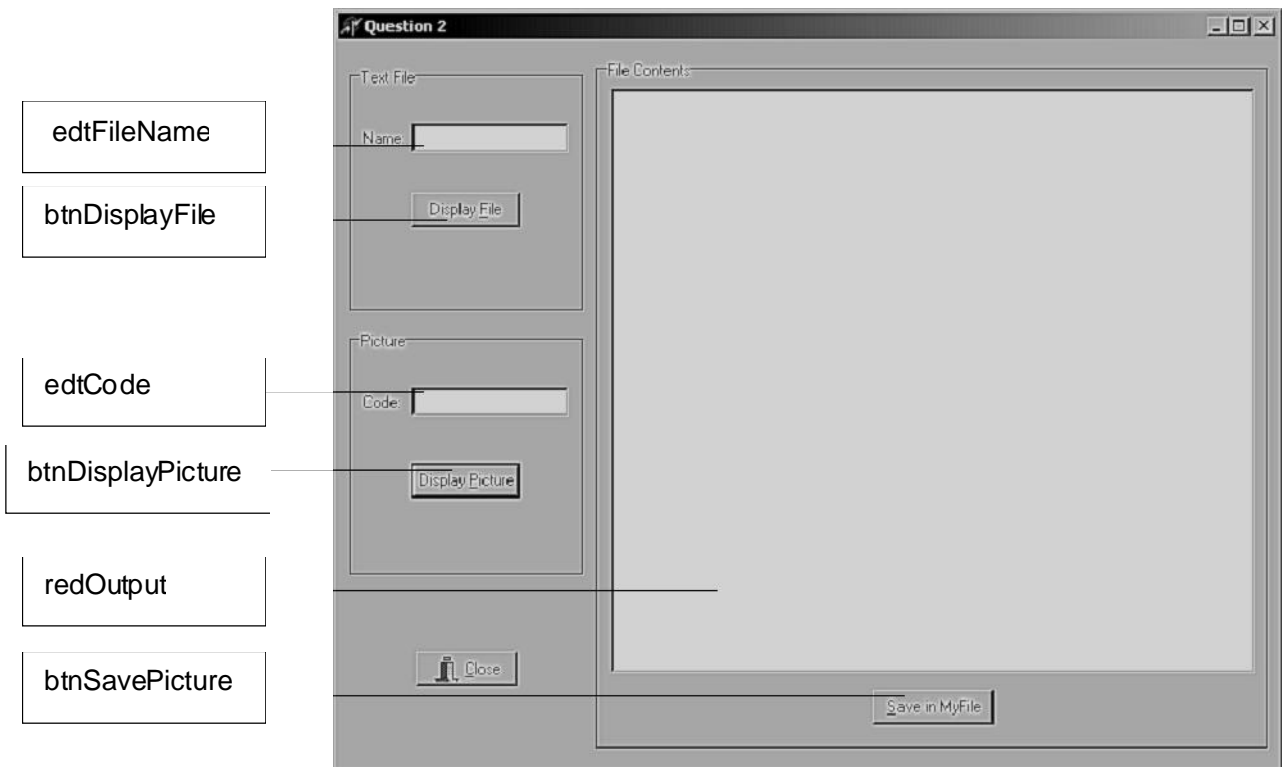
Delphi: (Question 2)

Open the file 'Ques2P.dpr' in Delphi, go to File|Save As... and save the *unit* as 'Ques2U_XXXX.pas' (XXXX represents the last four digits of your examination number). Now go to File|Save Project As... and save the *project* as 'Ques2P_XXXX.dpr'.

After you have answered Question 2.1 to 2.4, your program should be able to do the following:

- read the contents of 'mainfile.txt' and display it in a Rich Edit.
- extract a picture from 'mainfile.txt' and display it in the Rich Edit.
- write the picture that is displayed in the Rich Edit to a new text file.

2.1 Change **only** the *Caption* and *Name* properties of the different components on the form so that it corresponds with the figure below. Also add your examination number to the *Caption* of the form next to "Question 2". (2)



2.2 Write an event-handler for *btnDisplayFile* that will do the following:

- Whenever [Display File] is clicked, the contents of a specific text file must be displayed in *redOutput*.
- The user must type the name of the file that has to be displayed into *edtFileName*.

(8)

Test your program by displaying the contents of 'mainfile.txt'.

- 2.4 Write an event-handler for *btnSavePicture* that will do the following:
- Whenever [Save in MyFile] is clicked, the picture that is currently being displayed in *redOutput* must be written to 'MyFile.txt'.
 - Every time [Save in MyFile] is clicked, the new picture must be added **below** the previous picture in 'MyFile.txt'.

Choose any THREE pictures and save them to 'MyFile.txt'. (8)

The following must be handed in for Question 2 (Delphi):

- A printout of *Ques2U_XXXX.pas*.
- An "Alt | Print Scrn" while any picture is being displayed.

QUESTION 2: [50]

Pascal: (Question 2)

Start with a new program in Turbo Pascal and save it as 'XXXX_Q2.pas' (XXXX represents the last four digits of your examination number). The program must adhere to the following:

- 2.1 The menu below must be displayed **repeatedly** until the user chooses option 4:

1. Display the contents of a text file.
2. Display a picture.
3. Save picture to MyFile.txt
4. Quit.

(2)

- 2.2 For Option 1, write a procedure, *DisplayFile*, that will do the following:

- Whenever Option 1 is chosen, the contents of a specific text file must be displayed on screen.
- The program must first ask for the name of the file that should be displayed.

Test your program by displaying the contents of 'mainfile.txt'.

- 2.4 For Option 3, write a procedure, *SavePicture*, that will do the following:
- Whenever Option 3 is chosen, the last picture that was displayed must be written to 'MyFile.txt'.
 - Every time Option 3 is chosen, the new picture must be added **below** the previous picture in 'MyFile.txt'.

Choose any THREE pictures and save them to 'MyFile.txt'. (8)

The following must be handed in for Question 2 (Pascal):

- A printout of **XXXX_Q2.pas**.

QUESTION 2: [50]

TOTAL: 100

END