MODULE NAME:	MODULE CODE:
PROGRAMMING 2B	PROG6212

ASSESSMENT TYPE: ASSIGNMENT 1 (PAPER)

TOTAL MARK ALLOCATION: 100 MARKS

TIME ALLOWANCE: 15 HOURS

TOTAL PAGES: 8

TOPIC: LEARNING UNIT 1: DEBUGGING AND EXCEPTION HANDLING

STUDENT NAME:

STUDENT NUMBER:

INSTRUCTIONS:

1. *This is a practical assignment.*
2. *Assignments must be created using Microsoft Visual Studio 2010 and saved as appropriately named C# files.*
3. *Insert comments in your code which will identify you by name and student number.*
4. *All work must be adequately and correctly referenced. Code segments copied directly from a source must be referenced.*
5. ***Ensure that you follow good programming coding standards.***
6. ***This is a non-specific assignment; therefore, marks will be awarded for creativity and effort given.***
7. *This is an individual assignment. All work must be your own work.*
8. ***Make a copy/backup of your assignment before handing it in.***
9. ***No more than 25% of the assignment may be copied from original source(s) used, even if referenced correctly.***
10. *Follow all instructions on the assignment cover sheet.*

Assignment one has been set for you to demonstrate your C# programming abilities. You are required to use your knowledge of at least chapter one to chapter twelve in the text book. In other words, ensure that you demonstrate your understanding of: C#; methods, arrays, control structures; manipulation of strings; and the handling of user input, event based programming and advanced object-oriented programming (OOP) topics (inheritance and interfaces). Debugging and exception handling is a requirement.

Please note that when completing your assignment, you are required to use internationally acceptable coding standards (see page 112 in the prescribed textbook). Include comprehensive comments explaining variable names, methods and the logic of programming code. Document your application thoroughly. Although you are not required to submit user and developer type documentation with this assignment (marks will not be awarded for documentation), you are required to submit it with your Portfolio of Evidence and so it would make good sense to create this documentation while you are in the process of developing the application.

The types of system documentation you should at least consider are:

- Description of the system;
- System and program flowcharts;
- UML diagrams (at least class and use-case diagrams);
- Test data used to debug the application;
- Help files/ user guides for the system.

Application Specification (Marks: 100)

Before submitting your assignment ensure it meets the requirements set out in the marking rubric provided below.

1. Your application must make use of event handling and include debugging and exception handling.

2. Create a graphical user interface (GUI) login page for your project. The login page must:
    a. Allow the user to either register or login to the application. A first time user will need to register first before logging into the system.
    b. Accept a user name and password.

3. The application you develop should enable a user to apply at the Home Affairs department for biographical documentation (i.e. ID Book, Passport, Birth Certificate, marriage certificate, death certificate etc.). Your application should offer a selection of available documentation (minimum of 3 documents) that the user can apply for. Use Google™ to get examples of these documents and the types of information required for each document. The user will enter the additional information.

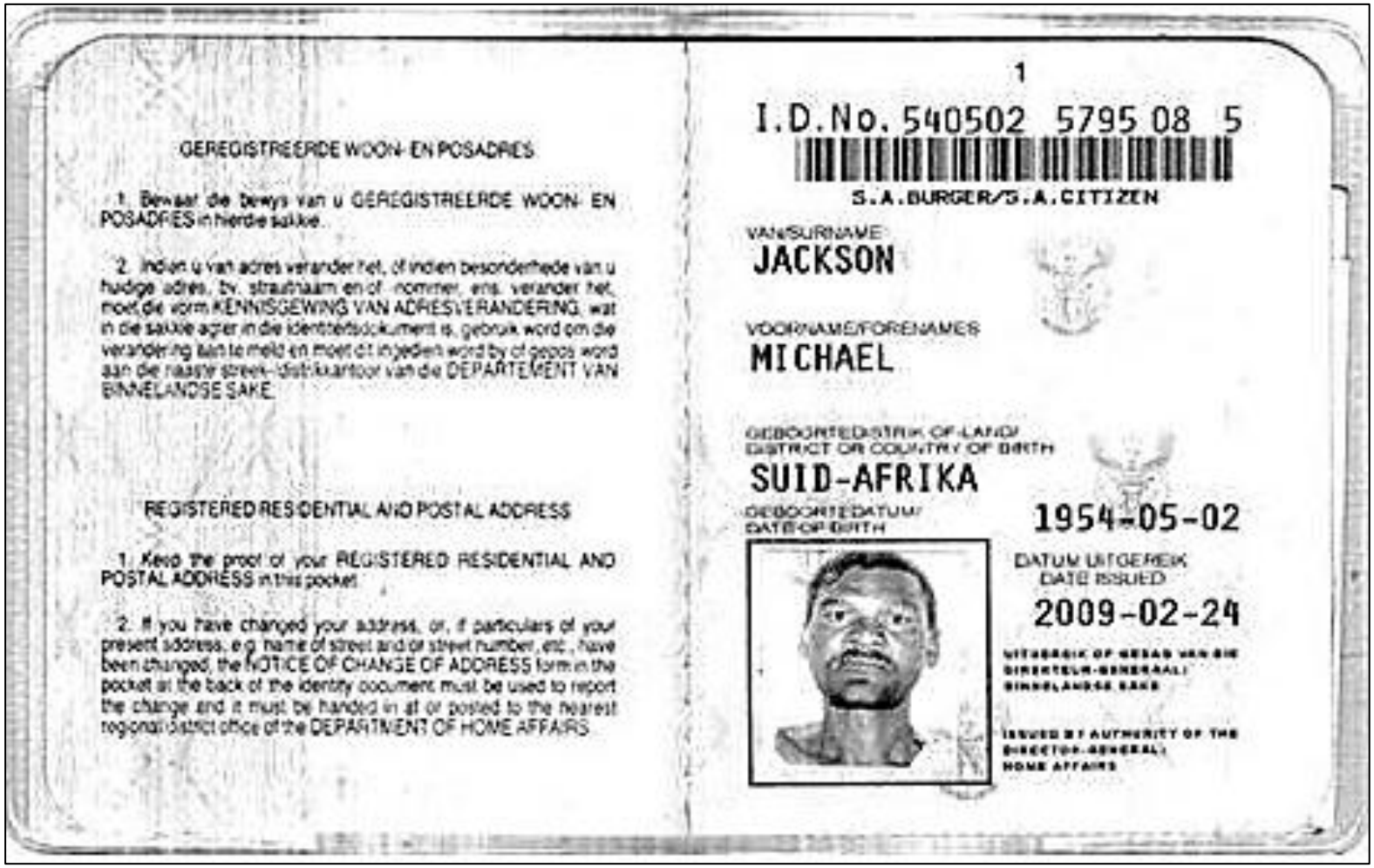
4. The base class will contain the most basic information about the user (Birth date, name and address) and the more document specific information will be stored in the derived (sub) classes.

5. Derive subclasses for the different documents you need to generate.

6. You will need to generate an ID number according to the following rules so you need to determine what data is input by the user. An example of C# programming code used to validate an ID number can be found here: http://www.dirkstrauss.com/programming/south-african-id-number-validation-in-c . Use it to determine how to generate an ID number. The rules for creating ID numbers are:
   - The first six (6) numbers are the birth date of the person in YYMMDD format;
   - The next four (4) are a gender, 5000 and above is male and below 5000 is female;
   - The next number is the country ID, 0 is South Africa and 1 is a foreigner;
   - The second last number used to be a racial identifier but now means nothing and is usually an eight (8);
   - The last number is a check bit. This verifies the rest of the number.

7. Ensure that the documents you generate contain all the information required in an official document issued by Home Affairs. Please note these official documents are not representative of real people (they are in actual fact fraudulent forged documents) E.g.

ID Document:



GEREGISTREERDE WOON- EN POSADRES

1. Bewaar die bewys van u GEREGISTREERDE WOON- EN POSADRES in hierdie sakkie.

2. Indien u van adres verander het, of indien besonderhede van u huidige adres, bv. straatnaam en of -nommer, ens. verander het, moet die vorm KENNISGEWING VAN ADRESVERANDERING, wat in die sakkie agter in die identiteitsdokument is, gebruik word om die verandering aan te meld en moet dit ingedien word by of gepos word aan die naaste streek-distrikkantoor van die DEPARTEMENT VAN BINNELANDSE SAKE.

REGISTERED RESIDENTIAL AND POSTAL ADDRESS

1. Keep the proof of your REGISTERED RESIDENTIAL AND POSTAL ADDRESS in this pocket.

2. If you have changed your address, or, if particulars of your present address, e.g. name of street and or street number, etc., have been changed, the NOTICE OF CHANGE OF ADDRESS form in the pocket at the back of the identity document must be used to report the change and it must be handed in at or posted to the nearest regional/district office of the DEPARTMENT OF HOME AFFAIRS.

1

I.D.No. 540502 5795 08 5

S.A.BURGER/S.A.CITIZEN

VAN/SURNAME
JACKSON

VOORNAME/FORENAMES
MICHAEL

GEBOORTEDISTRIK OF LAND/
DISTRICT OR COUNTRY OF BIRTH
SUID-AFRIKA

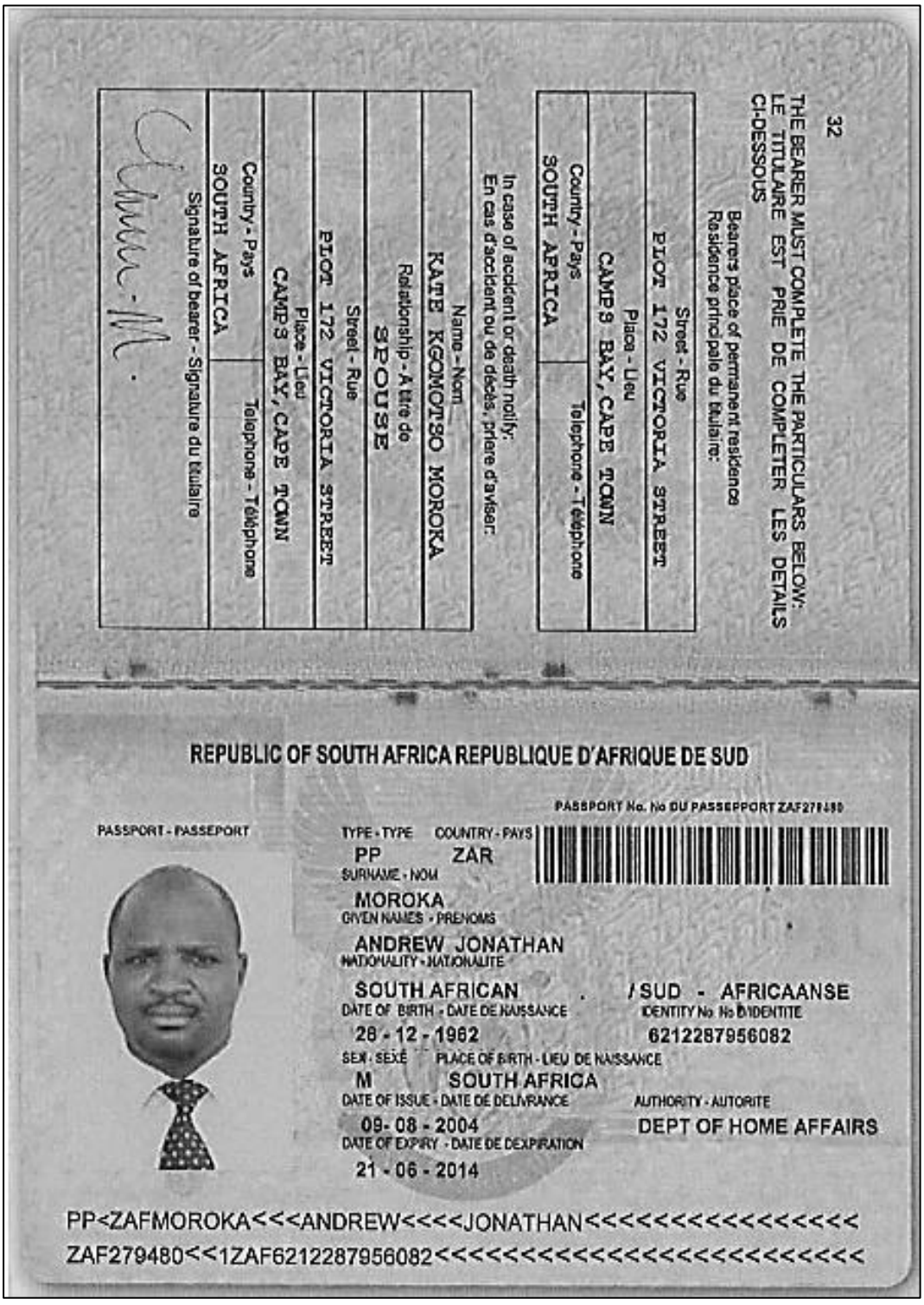
GEBOORTEDATUM/
DATE OF BIRTH
1954-05-02

DATUM UITGEREIK
DATE ISSUED
2009-02-24

UITGEREIK OP GESAG VAN DIE DIREKTEUR-GENERAAL: BINNELANDSE SAKE

ISSUED BY AUTHORITY OF THE DIRECTOR-GENERAL: HOME AFFAIRS

Passport:



8. Create options for at least three (3) biographical documents for the user:
    a. Once the user has logged in successfully, the user must have a choice of documents. The user should be provided with links to each document page. When a new user registers to use the application they need to enter their information to populate the base class (i.e. birth date, name and address). All the other information will be input once they have selected a document.
    b. Your application must include:
        i. User input and verification/ error handling;
        ii. Inheritance with constructors;
        iii. methods;
        iv. menus;
        v. Multiple decision statements;
        vi. Loops;
        vii. String manipulation; and
        viii. Regular expressions;
        ix. Debugging and exception handling.

9. The outline of this assignment is broad. A guideline has been provided for you to work within but the look and feel of your application will differ from those your peers submit. Use your imagination. Add additional functionality to your assignment where necessary.

10. Ensure that you have covered all items listed in the marking rubric below. The mark assigned will follow the following sliding scale:

0 Mark	-	Not completed / not submitted;
1 Mark	-	Attempted but incorrect;
2 Marks	-	Attempted, logic / process in the correct direction;
3 Marks	-	Logic / process correct;
4 Marks	-	Logic / process and coding correct;
5 Marks		Excellent, Logic / process correct and coding correct and efficient. Student has gone above and beyond the specified requirements.

Marking Guidelines

Criteria	Possible mark	Mark awarded
Good coding standards • Variable naming; • Comments / code readability throughout C# code; • C# file naming (modular coding – using different C# files for different modules); and • User friendly design; • Efficient code (no redundancy).	5 5 5 5 5	
Subtotal	25	
Login page • Registration of a new user • Accepts user name and password; and • Validation of login and password	5 5 5	
Subtotal	15	
Application page(s) • Choice of at least 3 biographical documents • User input of details; • Inheritance; • Customised class/ static methods; • Menus; • Multiple decision statements; • Loops; • String manipulation; and regular expressions; • Debugging/ exception handling/Error handling/ validation; • Output.	5 5 5 5 5 5 5 5 5 5	
Subtotal	50	

Other Marks • Student has added extra functionality to the application for the user; • Student has used advanced features not covered in class.	5 5	
Subtotal	10	
Total	100	

[TOTAL MARKS: 100]