FACULTY OF SCIENCE

ACADEMY OF COMPUTER SCIENCE AND SOFTWARE ENGINEERING

MODULE	**IFM01A1 & IFM1A10** INFORMATICS 1A: INTRODUCTION TO ALGORITHM DEVELOPMENT (VB)
CAMPUS	APK
EXAMINATION	JUNE 2019 – PAPER A

DATE	2019-06-11	**TIME**	12:30–15:30
ASSESSORS		K LEBEA WS LEUNG	
INTERNAL MODERATOR		HJC VAN DER WESTHUIZEN	
DURATION	3 Hours	**MARKS**	100

PLEASE READ THROUGH THESE INSTRUCTIONS CAREFULLY AND ADHERE TO THEM

The Academy takes NO responsibility for your ignorance (willful or otherwise) and will NOT grant additional opportunities to students who fail to submit their work correctly

☞ This question paper comprises 4 (four) pages (including this cover page)
☞ The use of calculators are NOT permitted during this examination
☞ The first 20 (twenty) minutes are reserved for DESIGN (must be done in answer book provided)
☞ Answer ALL questions
☞ No further marks will be awarded for Correct Execution from the point your submitted program terminates unexpectedly

☞ Follow the project naming convention indicated on your number card.

☞ SAVE REGULARLY - no additional time will be granted due to power failures
☞ Save ONLY to the T:\ directory
☞ Do NOT save to other locations outside of the default project folder - use ONLY the Save All button

☞ Read through page 1 for submission guidelines when you are ready to submit. Be sure to complete your details
☞ An invigilator must be alerted to verify and take in your submission

DO NOT TURN OVER THIS QUESTION PAPER UNTIL YOU HAVE BEEN GIVEN INSTRUCTION TO COMMENCE

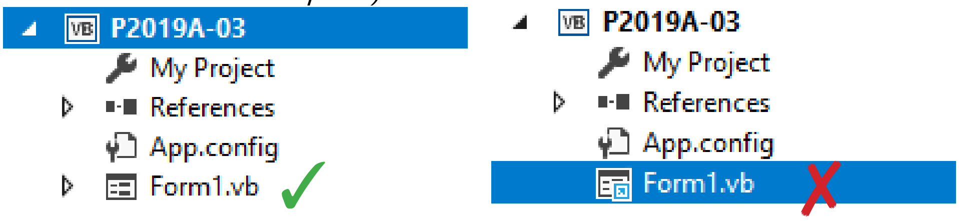
When you are ready to submit your practical...

a) On your exam booklet, CLEARLY indicate your details, INCLUDING your Computer Number

b) Preparing your submission
- Zip up the folder containing your project - ensure that this zip contains your entire project

c) Uploading to Eve
- Visit http://eve.uj.ac.za
- Select File Storage from the Menu
- Select the Session Informatics 1A Examination
- Enter your username (your student number)
- Enter the password (found on the back of your number card placed at your computer
- Upload the Zip file you created in Step b)

d) Verifying the Upload
- Download the file you uploaded to another location
- Unzip it to verify that the entire project is there *(make sure that your project does not show any external links in the Solution Explorer)*



P2019A-03
My Project
References
App.config
Form1.vb ✓

P2019A-03
My Project
References
App.config
Form1.vb ✗

e) Preparing the Backup to CD (use the CD provided to you)
- Burn EVERYTHING on the T:\ drive to the CD
- Use your *Student Number* for the **Disc title** and Select **With a CD/DVD player**
- If your CD fails, alert an assistant to backup your project to USB instead

f) Agreement and Submission ***(to be completed when you submit)***
- Complete your details - by signing below, you acknowledge that you have done the above steps to ensure that your submission is correct.

Checklist ***(check all boxes)***:

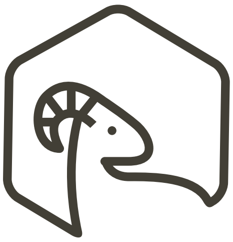
❑ I have followed the naming convention as indicated on my password card
❑ A zip of the full and final project has been uploaded to Eve
❑ A backup of the full and final project has been made, either to CD or USB

I further acknowledge that:

❑ I am aware that a non-compiling submission is capped at 40%
❑ Code I comment out will not be marked
❑ I take full responsibility for ensuring that my submission is correct and the version I intend to submit for assessment

Student #										PC #			
ID/Passport #													
Signature													
Backup Used	CD							USB					

Hand everything in to the invigilators *(this question paper, the assessment script, CD, number card).*



The final season of GoaTy *(a series about rambunctious goats who battle each other so that they can take the top spot at the top of a hill)* is finally winding down with the penultimate episode having aired last weekend.

The executives of the studio that produces GoaTy are naturally pleased with all the money raking in. However, they are concerned that GoaTy has been so popular that Internet pirates are well...pirating the episodes! They are pretty annoyed and have approached you huffily and puffily, requesting that you design and develop a Visual Basic application that will help them track down the number of times GoaTy is being downloaded *(illegally)*.

0.1	Design *(Input, Output, Events & Actions, Variables, Record structures, Interface, Test Data, Algorithms)*	5
0.2	Implementation of Interface	2
0.3	Option Statements	1
0.4	Effective use of subroutines / functions	1
0.5	Commenting	1

Here are the requirements that they have specified:

1) The application must be able to store the following information for each season of GoaTy:

Field	Example Data
a) Season Number	1
b) Number of Executive Producers on Season	6
c) Data for each Episode in the Season:	
i) Title of the Episode	`"The Wolves!"`
ii) Minutes	52
iii) Times Episode Pirated in each Country	125; 265; 41; ...; 2621
iv) Total Times Episodes Pirated	See Question 4
d) Most Pirated Episode of the Season	See Question 5
e) Season Rating	See Question 7

1.1	Definition of Episode Type record structure	4
1.2	Definition of Season record structure	5

2) The application must be able to handle any number of Seasons, any number of Episodes per Season, and any number of Countries monitored for the Episode. These numbers are specified by the user at the beginning. For some bizarre reason *(which is convenient to you)*, the producers of GoaTy decided that each Season will have the same number of episodes. They are also going to monitor the same number of countries for each episode.

2.1	Input of number of Seasons, Episodes, and Countries	2
2.2	Set up of dynamic arrays	5
2.3	Set up of UJGrid control	2
2.4	Labelling of UJGrid control headings *(Seasons as Rows, and Episodes as Columns)*	3

3) The user must be able to input the necessary data into the program.

3.1	Input of data for each Season *(including data on each Episode)*	5

4) For each Episode of each Season, calculate, store *(in field b-iv)*, and display the total times episodes were downloaded illegally.

4.1	Calculation of total times episodes were downloaded	4
4.2	Display of calculation of total times episodes were downloaded	2
4.3	Correct	6

5) For each Season, calculate, store *(in field d)* the index of the Episode that was the most downloaded. Use the index to display the titles of these episodes.

5.1	Calculation of index of the most illegally downloaded episode per season	5
5.2	Display of title of most illegally downloaded episode	1
5.3	Correct	6

6) Define a subroutine called `ConvertToHours` which will convert minutes into hours and minutes. For example, 110 minutes will return the Integer values 1 *(hours)* and 50 *(free minutes)*. The subroutine will have a total of three Integer parameters.

6.1	Definition of ConvertToHours subroutine	2
6.2	Calculation of Hours and Free Minutes based on Minutes	3

7) For each Season, store *(in field e)*, and display the Season Rating. This must be calculated using one `If Statement` and two `Select Case` statements to evaluate the following:

Number of free minutes	Hours = 0	Hours > 0
0	1	2
1 to 30	3	4
31 to 59	5	6

7.1	Calculate Season Ratings based on Free Minutes and Hours of Most Pirated Episode *(using `ConvertToHours` subroutine)*	4
7.2	Display of Average of Season Ratings	1
7.3	Correct	10

8) The studio would like to know if the number of producers is increasing by the season. For example, if Season 1 has 1 producer, Season 2 has 2 producers, and Season 3 has 3 producers, then it is increasing. If it is 1, 1, 2, then it is not.

8.1	Determine if increasing trend exists from Season to Season	8
8.2	Display	2
8.3	Correct	10

＼(＾∀＾)メ(＾∀＾)ノ