



UNIT 3

PRACTICAL PROGRAMMING

PROJECT

DESIGN LINKED TO DEVELOPMENT

PROJECT OVERALL



• ANALYSIS	10	
• DESIGN	15	
• DEVELOPMENT	25	
• EVALUATION	20	
• TOTAL	70	
• VALUE		20%

COMPONENTS




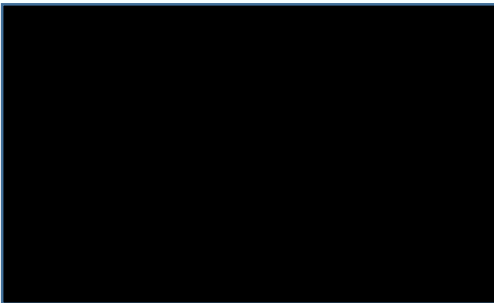
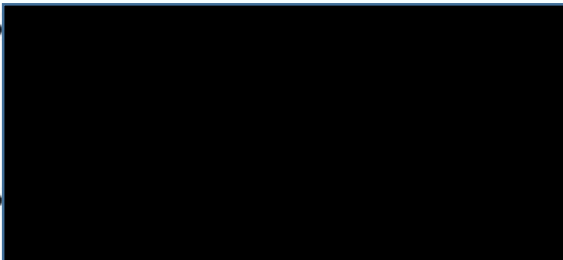
3.2 Design of the solution (15 marks)	
3.2.1 Decompose the problem	(a) Break down the problem into smaller parts suitable for computational solutions justifying any decisions made.
3.2.2 Describe the solution	(a) Explain and justify the structure of the solution. (b) Describe the parts of the solution using algorithms justifying how these algorithms form a complete solution to the problem. (c) Describe usability features to be included in the solution. (d) Identify key variables / data structures / classes justifying choices and any necessary validation.
3.2.3 Describe the approach to testing	(a) Identify the test data to be used during the iterative development and post development phases and justify the choice of this test data.

3.2.2

DESCRIBE THE PROBLEM



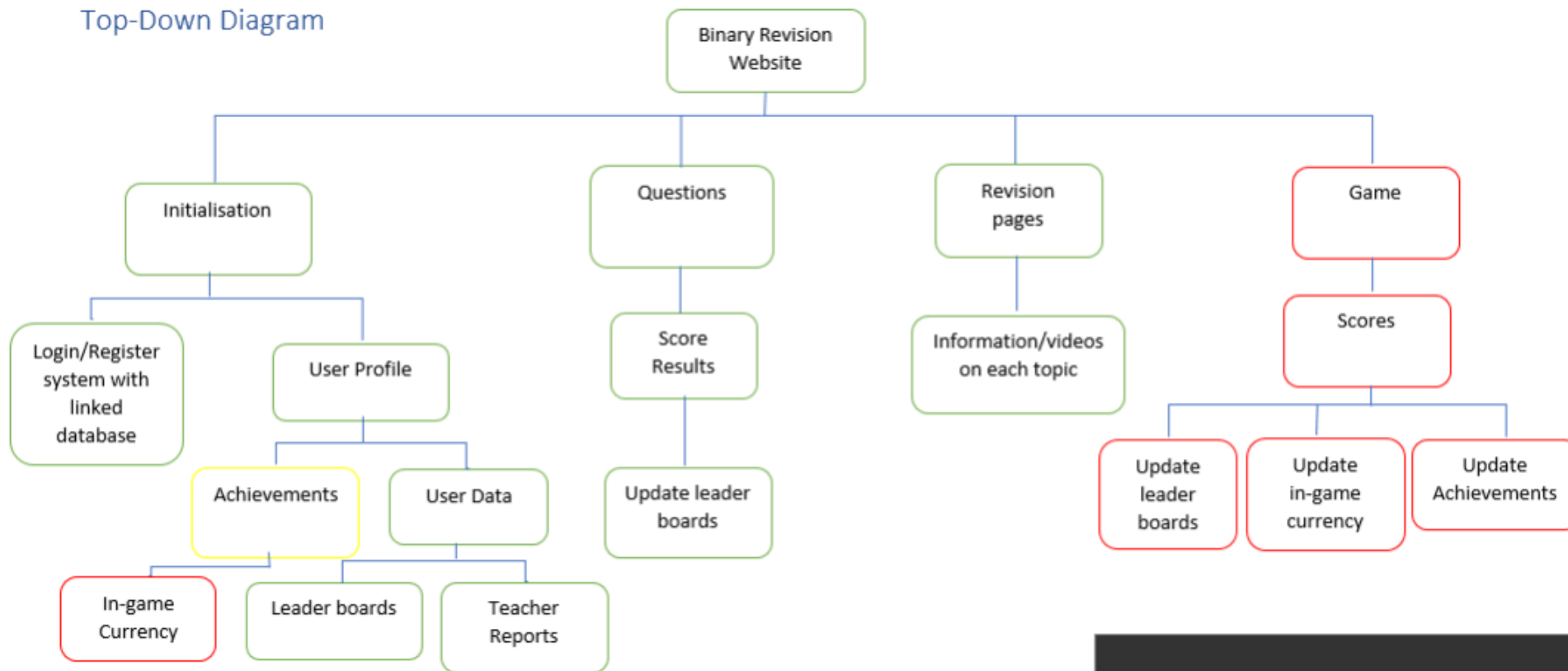
(b) Describe the parts of the solution using algorithms justifying how these algorithms form a complete solution to the problem.

AO 3.1 Design (maximum 15 marks)			
1–4 marks	5–8 marks	9–12 marks	13–15 marks
The candidate will have:			
<ul style="list-style-type: none">Described elements of the solution using algorithms.	<ul style="list-style-type: none">Described the solution fully using appropriate and accurate algorithms.	<ul style="list-style-type: none">Described the solution fully using appropriate and accurate algorithms explaining how these algorithms form a complete solution to the problem.	<ul style="list-style-type: none">Described the solution fully using appropriate and accurate algorithms justifying how these algorithms form a complete solution to the problem.

TOP DOWN DESIGN



Top-Down Diagram



NVIDIA Control Panel is not found

JUSTIFICATION



- WHY USE DECOMPOSITION IN THIS WAY:
- **EG**
- **BINARY REVISION WEBSITE**
- THIS IS THE OVERALL PRODUCT THAT I WILL BE PRODUCING, WHICH CAN BE BROKEN DOWN INTO MANY SMALLER SUB-PROBLEMS USING DECOMPOSITION. THESE SUB-PROBLEMS CAN THEN BE SOLVED INDIVIDUALLY USING SEPARATE FUNCTIONS FOR MOST OF THEM, THIS MEANS THAT I CANNOT INTERRUPT THE LOGIC OF A SUB-PROBLEM WITH ANOTHER, MEANING THAT IT IS MUCH LESS LIKELY FOR ME TO CAUSE AN ERROR IN A SEPARATE FUNCTION.

JUSTIFICATION



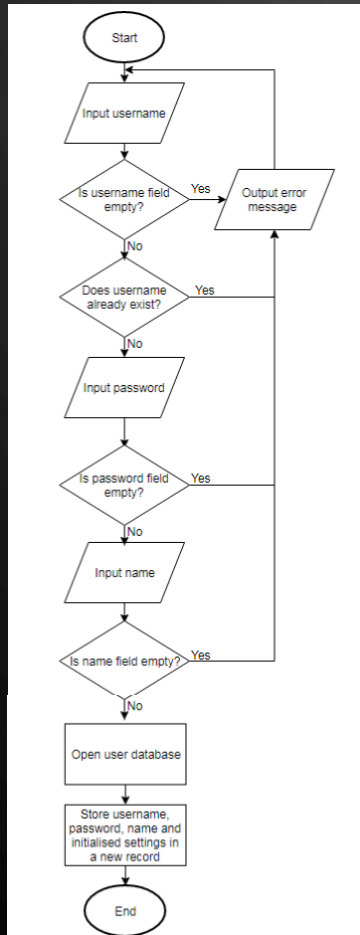
- WHAT DOES EACH ALGORITHM DO?
- **LOGIN/REGISTER SYSTEM**
- THIS WILL BE AN INDIVIDUAL MODULE SO THAT IT CAN EASILY BE TESTED IN ISOLATION AND NOT AFFECT ANY OTHER AREAS OF THE PROGRAM, THIS WILL ENSURE THAT I CAN IDENTIFY ANY ISSUES WITHIN THE FUNCTION.
- THE LOGIN/REGISTERING SYSTEM WILL CONTAIN 2 BUTTONS FOR THE USER TO CLICK ON, EITHER LOGIN OR REGISTER. IF THE USER CLICKS LOGIN, THEN THEY SHALL BE ASKED TO INPUT THEIR USERNAME AND PASSWORD. IF THIS IS CORRECT THEN THAT CAN CONTINUE TO THE REST OF THE WEBSITE WHICH WILL BE CHECKED BY REVIEWING THE DATABASE TO SEE IF THEY MATCH, HOWEVER IF THIS IS INCORRECT AN ERROR SHALL APPEAR AND THEY WILL BE ASKED TO RE-ENTER THEIR DETAILS. IF THEY CLICKED ON REGISTER INSTEAD THEN THEY SHALL BE ASKED TO INPUT THEIR USERNAME, EMAIL, PASSWORD, AND THE PASSWORD AGAIN FOR VALIDATION. IF THE USER INPUTS A TAKEN USERNAME OR EMAIL THEN THEY SHALL BE ASKED TO ENTER A DIFFERENT ONE, THIS WILL BE CHECKED BY SCANNING THE DATABASE TO SEE IF THEY EXIST. ALSO, IF THE PASSWORDS DO NOT MATCH OR IS TOO SHORT, THEY WILL BE ASKED TO ENTER THE PASSWORD AGAIN THAT MEETS THE CRITERIA. ONCE THE USER HAS SUCCESSFULLY LOGGED IN THEY SHALL THEN BE SHOWN TO THE NEXT SECTION, USER PROFILE.

WHAT DOES THIS MEAN?



- TAKE EACH PART OF THE TOP DOWN DIAGRAM AND CREATE THE ALGORITHMS USING FLOWCHARTS AND PSEUDOCODE.

EXAMPLE CREATING AN ACCOUNT



When the user creates an account they need to enter their name, username and password. An ID needs to be assigned to them and all other fields need to be initialised with a dash, -, symbol.

EXAMPLE CREATING AN ACCOUNT



```
01 function create_account(username,password,name)
02     valid_1 = check_username(username)
03     valid_2 = check_existence(password)
04     valid_3 = check_existence(name)
05     if (valid_1 = False) or (valid_2 = False) or (valid_3 = False) then
06         return False #failed to create account
07     endif
08     file = openRead("User Database") #read only mode
09     n = 0
10     while NOT file.endOfFile()
11         n += 1
12     endwhile
13     file.close()
14     file = openWrite("User Database") #append mode would add the record at end of file
15     file.writeLine(n+",",name+",",username+",",password+",", "-"+", "-"+", "+"\n")
16     file.close()
17     return True #account created
18 endfunction
```

From the flowchart I then used pseudocode to plan out three functions that will be needed in the process of creating an account.

The function below calls for other functions to check whether the username, password and name are valid. If they are valid then it writes the user's details into the user database and initialises the other fields. If the username, password or name aren't valid then it returns False.

3.2.3

DESCRIBE THE APPROACH TO TESTING



(a) Identify the test data to be used during the iterative development and post development phases and justify the choice of this test data.

AO 3.1 Design (maximum 15 marks)			
1–4 marks	5–8 marks	9–12 marks	13–15 marks
<ul style="list-style-type: none">Identified some test data to be used during the iterative or post development phase of the process.	<ul style="list-style-type: none">Identified the test data to be used during the iterative development of the solution.	<ul style="list-style-type: none">Identified and justified the test data to be used during the iterative development of the solution.	<ul style="list-style-type: none">Identified and justified the test data to be used during the iterative development of the solution.

WHAT DOES THIS MEAN?



- COMPLETE A TRACE TABLE FOR EACH TO TEST THAT THE LOGIC WITHIN THE PSEUDOCODE IS WORKING.
- MAKE IT CLEAR WHICH TEST DATA IS BEING USED, EXPECTED OUTPUT AND ACTUAL OUTPUT FOR EACH ALGORITHM.
- ALSO MAKE IT CLEAR THAT THIS DATA WILL BE USED THROUGHOUT THE DEVELOPMENTAL STAGES.

TRACE TABLE

Trace Table

To test the logic within the pseudocode I have completed trace tables for different sets of data.

Test 1

The first test uses normal data. Test Data:

Username	Password	Name
red	pass	Rosie
#doesn't already exist		

Trace Table

Line No.	Username	Password	Name	valid_1	valid_2	valid_3	<u>not_empty</u>	Output
01	red	pass	Rosie					
42								Function <u>check_existence</u> returns True
20							True	
35								Function <u>check_username</u> returns True
02				True				
42								Function <u>check_existence</u> returns True
03					True			
42								Function <u>check_existence</u> returns True
04						True		
17								Function <u>create_account</u> returns True



TRACE TABLE

Test 2

The second test uses erroneous data. Test data:

Username	Password	Name
red	pass	
#exists		#empty field

Line No.	Username	Password	Name	valid_1	valid_2	valid_3	<u>not_empty</u>	Output
01	red	pass						
42								Function <u>check_existence</u> returns True
20							True	
29								Display "Error. Username already exists"
31								Function <u>check_username</u> returns False
02				False				
42								Function <u>check_existence</u> returns True
03					True			
39								Display empty field message
40								Function <u>check_existence</u> returns False
04						False		
06								Function <u>create_account</u> returns False

In both tests, the algorithm works correctly.



EVIDENCE



- ALGORITHMS FOR EACH KEY ASPECT GIVEN IN THE DECOMPOSITION PHASE
- EXPLANATION OF WHAT THE ALGORITHM DOES-
 - IS IT LINKED TO OTHERS?
- ALGORITHM IN THE FORM OF A FLOWCHART AND PSEUDOCODE
- TRACE TABLES USING EXPLICIT DATA FOR NORMAL, EXTREME AND ERRONEOUS VALUES

COMPONENTS



3.3 Developing the solution (25 marks)	
3.3.1 Iterative development process	<p>(a) Provide annotated evidence of each stage of the iterative development process justifying any decision made.</p> <p>(b) Provide annotated evidence of prototype solutions justifying any decision made.</p>
3.3.2 Testing to inform development	<p>(a) Provide annotated evidence for testing at each stage justifying the reason for the test.</p> <p>(b) Provide annotated evidence of any remedial actions taken justifying the decision made.</p>

3.3.1

ITERATIVE DEVELOPMENT PROCESS



(a) Provide annotated evidence of each stage of the iterative development process justifying any decision made.

AO 3.2 Developing the coded solution (maximum 25 marks)			
Iterative development of a coded solution (maximum 15 marks)			
1–4 marks	5–8 marks	9–12 marks	13–15 marks
The candidate will have:			
<ul style="list-style-type: none">• Provided evidence of some iterative development for a coded solution.• Solution may be linear.• Code may be inefficient.	<ul style="list-style-type: none">• Provided evidence for most stages of the iterative development process for a coded solution describing what they did at each stage.• Solution will have some structure.	<ul style="list-style-type: none">• Provided evidence of each stage of the iterative development process for a coded solution relating this to the break down of the problem from the analysis stage and explaining what they did at each stage.	<ul style="list-style-type: none">• Provided evidence of each stage of the iterative development process for a coded solution relating this to the break down of the problem from the analysis stage and explaining what they did and justifying why.• Provided evidence of prototype versions of their

WHAT DOES THIS MEAN?

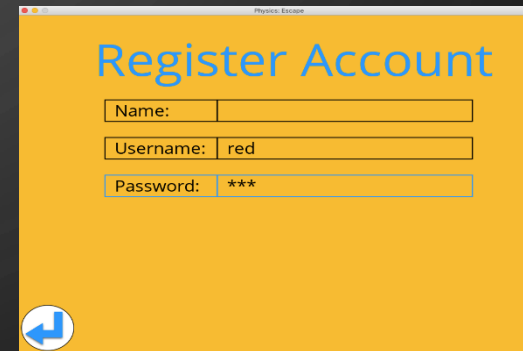


- FOR EACH ALGORITHM SHOW STEPS INVOLVED IN CODING THE SOLUTION.
 - ANY PROBLEMS ENCOUNTERED DOCUMENT CODE EXPLAINING HOW IT WAS DEBUGGED
 - PROOF OF IT RUNNING
 - RESEARCH CARRIED OUT (WITH SOURCES) ETC
 - GIVE ANNOTATED CODE AT EACH STAGE

Further validation will be required for the name as names only contain letters. But this will be done when the user attempts to register an account.

The next aim is to ensure that anything typed in the password field will appear as asterisks. It would require the `update(mouse_pos, chosen, text)` method to determine whether the field is the password field. I've used a local variable `text` to store what will be displayed. This way the attribute `detail` is unaffected.

```
if self.num == 3:
    text = "*" * len(self.detail)
else:
    text = self.detail
#display user detail
x_pos = self.pos_x+self.label_width+20
print_text(self.font,x_pos,self.pos_y,text,self.text_colour)
```



Register Account

Name:	<input type="text"/>
Username:	red
Password:	***



3.3.2

TESTING TO INFORM DEVELOPMENT



- (a) Provide annotated evidence for testing at each stage justifying the reason for the test.
- (b) Provide annotated evidence of any remedial actions taken justifying the decision made.

Testing to inform development (maximum 10 marks)			
1–2 marks	3–5 marks	6–8 marks	9–10 marks
The candidate will have:			
<ul style="list-style-type: none">• Provided some evidence of testing during the iterative development process.	<ul style="list-style-type: none">• Provided some evidence of testing during the iterative development process.• Provided evidence of some failed tests and the remedial actions taken.	<ul style="list-style-type: none">• Provided evidence of testing at most stages of the iterative development process.• Provided evidence of some failed tests and the remedial actions taken with some explanation of the actions taken.	<ul style="list-style-type: none">• Provided evidence of testing at each stage of the iterative development process.• Provided evidence of any failed tests and the remedial actions taken with full justification for any actions taken.

WHAT DOES THIS MEAN?



- FOR EACH ALGORITHM TESTING OCCURS DURING DEVELOPMENT.
 - USE THE TEST DATA AND RESULTS GAINED IN THE TRACE TABLES
 - PROOF OF IT RUNNING
 - COMMENTS ON SUCCESS AND/OR REMEDIAL ACTIONS REQUIRED
 - GIVE ANNOTATED CODE AT EACH STAGE

TEST 2:

THE EXPECTED OUTCOME IS THAT THE MESSAGE "ERROR. EMPTY FIELD" IS DISPLAYED. THIS IS BECAUSE THE PRESENCE CHECK IS THE FIRST VALIDATION THAT OCCURS. ACTUAL OUTCOME:



Username	Password	Name
red	pass	
#exists		#empty field

A screenshot of a web browser window titled "Physics: Escape" showing a "Register Account" form. The form has three input fields: "Name:" (empty), "Username:" (containing "red"), and "Password:" (containing "****"). A blue "Register" button is positioned below the fields. Below the button, the text "Error. Empty Field." is displayed. In the bottom-left corner of the form area, there is a circular icon with a blue arrow pointing left.

GOOD LUCK!



- AIM TO GET 1 ALGORITHM DONE AS THIS!!
- DON'T CODE AND FORGET TO DOCUMENT THIS SECTION!
- CONTACT ME WITH ANY CONCERNS!

Development

```
Please select an option?
Denary to Binary
Enter a number?5
Traceback (most recent call last):
  File "F:\the big one.py", line 442, in <module>
    mainMenu()
  File "F:\the big one.py", line 20, in mainMenu
    denaryMenu()
  File "F:\the big one.py", line 50, in denaryMenu
    DenBin()
  File "F:\the big one.py", line 357, in DenBin
    ar[0]=()
NameError: name 'ar' is not defined
>>>
```

```
#denary to binary
def DenBin():
    den=128
    ar=["", "", "", "", "", "", "", "", ""]
    if den>128:
        den=den-128
```

The list "ar" needed to be defined with empty slots for the new values to be put into so that the values became individual (see Iterative test table)

The code didn't work as it was missing speech marks around the circled areas below, meaning it treated it as an integer rather than string which lead to the program not being able to compare the variable to the character. (See Iterative test table/test 5)

```
if num[7]==1:
    print("1 turned on")
    total=total+1
elif num[7]==0:
    print("1 turned off")
else:
```

```
Please select an option?
Binary to Denary
This will show you how to convert from binary to denary

Enter an 8 bit binary number: 00001010
correct
128 turned off
64 turned off
32 turned off
16 turned off
8 turned on
4 turned off
2 turned on
Invalid input try again
This will show you how to convert from binary to denary

Enter an 8 bit binary number:
```



Figure 12

I created this section of code, it still came up with the same result as last time and comes up with a name error. And this same error occurred when I started creating my login system.



```
#Next is the Login window design

def SignIn():
    global sign-in_screen
    sign-in_screen = Toplevel(sign-in_screen)
    sign-in_screen.title("Sign-In")
    sign-in_screen.geometry("300x250")
    Label(sign-in_screen, text="Please enter details below to login").pack()
    Label(sign-in_screen, text="").pack()

    global verify_username
    global verify_password

    verify_username = str()
    verify_password = str()

    global username_sign-in_entry
    global password_sign-in_entry

    Label(sign-in_screen, text="Username = ").pack()
    username_sign-in_entry = Entry(sign-in_screen, textvariable=verify_username)
    username_sign-in_entry.pack()
    Label(sign-in_screen, text="").pack()
    Label(sign-in_screen, text="Password = ").pack()
    password_sign-in_entry = Entry(sign-in_screen, textvariable=verify_password, show='*')
    password_sign-in_entry.pack()
    Label(sign-in_screen, text="").pack()
    Button(sign-in_screen, text="Login", width=10, height=1, command = login_verify).pack()
```

Figure 13

```
RESTART: /Users/johngallego/Desktop/JK Sixth Form Work/Computer Science JK/Python Computer Science/Login System
Traceback (most recent call last):
  File "/Users/johngallego/Desktop/JK Sixth Form Work/Computer Science JK/Python Computer Science/Login System Tkinter 2.py", line 85, in <module>
    Label(text="").pack()
NameError: name "text" is not defined. Did you mean: "Text"?
```

Figure 14



GRADE BOUNDARIES COURSEWORK



Grade	Score /70
A*	63
A	55
B	47
C	40
D	33
E	26