Level 4 Diploma in Computing

Software Development Techniques

Candidates Name	**: Fatema Akter**
NCC Candidate No	**: 00154713**
Title	**: Computer Networks**

Candidate attempting to gain an unfair advantage or colluding in anyway whatsoever (other than on joint assignments) are liable to be disqualified. Plagiarism is an offence.

Mark	Moderated	Final
	Mark	**Mark**

Marker's comment:

Expected candidate time allocation: 35 to 40 hours

Statement of confirmation of own work

Programmed /qualification name: Software Development Techniques

Student declaration:

I have read and understood NCC Education's policy on Academic Dishonesty and Plagiarism.

I can confirm the following details**:**

Student ID/Registration number	**: 00154713**
Name	**: Fatema Akter**
Centre Name	**: Daffodil Institute of IT**
Module Name	**: Software Development Techniques**
Assignment title	**: Software Development Techniques**
Number of Words	**: 1976**

I confirm that this is my own and that I have not plagiarized any part of it. I have also noted the assessment criteria and pass mark for assignments.

Due Date	**:**
Student Signature	**:**
Submitted Date	**:**

Contents

Introduction:

We must develop a simple arithmetic tester program can be used to help young children improve their arithmetic skills. Our arithmetic program should be a simple program which accepts string data for the player's name and the number of questions they want to try, which will be entered by the user. Our program must take in the following inputs

from the user. The program should generate one question at a time and display the question for the user. The questions must involve two numbers and be a multiplication, division, subtraction, or addition. Now I have to submit a detailed report of my program.

Task -1

Function for player name validation

1. Function playerName (needs name as string) returning name
2. If name less than 2 and grater 20
3. Output “data player name is invalid”
4. End if
5. Else
6. Output “data player name is valid”
7. Return name
8. End function

Function for question number validation

1. function question number(needs quesnumber as whole number) returning whole number
2. If number less than 2 and grater 50
3. Output “ data question number is invalid”
4. End if
5. Else
6. Output “data question number is valid”
7. End function

Function question generator

1. Function question renerator(needs first num array[]as whole number, last num[] as whole number)returing whole number
2. loop
3. Data data first number as whole number
4. Data last as whole number
5. If question less than 1and grater 7
6. Return true
7. End if

8. end loop
9. end function

Function for addition

1. function addition (needs Firstnum,lastnum)
2. Data result as whole number
3. Call numbergenerator()
4. Result=fristnum + lastnum
5. Return result
6. End function

Function for subtraction

1. function subtraction (needs Firstnum, lastnum)
2. Data result as whole number
3. Call numbergenarator()
4. Result= fristnum - lastnum
5. Return result
6. End function

Function for division

1. function division(needs Firstnum,lastnum)
2. Data result as whole number
3. Call numbergenerator()
4. Result= fristnum / lastnum
5. Return result
6. End function

Function for multiplication

1. function multipliction (needs Firstnum , lastnum)

2. Data result as whole number
3. Call numbergenerator()
4. Result= fristnum * lastnum
5. Return result
6. End function

Function for Random generator

1. Function numbergenarator () returns ##########
2. Data firstnum as whole number
3. Data lastnum as whole number
4. firstnum = rand(0,12)
5. lastnum = rand(0,12
6. return firstnum
7. return lastnum
8. end function

Function for Question stores

1. function question store (needs arrayfirst number[] and arraylast number[] as whole numbers , needs first num and last numas whole number)
2. Data first number as whole number
3. Data last number as whole number
4. loop while first number less than 2 and grater or equal to 50 and
5. last number less than 2 and grater or equal to 50
6. if arrayfirst num[first number] less than arraylast num[last number]
7. else if arraylast num[last number] less than arrayfirst num [first num])
8. else
9. if arrayfirst num[first number] summation of arraylast num[last number]
10. loop while
11. output “Please input the question store you found”
12. end if
13. end loop
14. End function

Function for Match result

1. function match result (needs first num as whole number, last num as whole number,rest as whole number) returning whole number
2. data firstnumber as whole number

3. data lastnumber as whole number
4. data result as whole number
5. data counter as whole number
6. output' please enter first number"
7. input firstnumber
8. output' please enter last number"
9. input lastnumber
10. result=first number
11. counter=1
12. counter=last number
13. result=result+first number
14. counter=counter+1
15. next loop
16. end function

Function for wrong answer

1. function wronganswer (needs wrgans whole number) returning whole number
2. if data number less than 2 and grater 20
3. return true
4. end if
5. else
6. return false
7. output "please enter the valid answer"
8. end function

Task- 2

Main function for (Pseudo code)

```
1    Data name as string
2    Data number as whole number
3    Output “please enter the name”
4    Input name
5    Output “please enter the number”
6    Input number
7    data answers as whole number
8    data validation as whole number
9     output “please enter the answers”
10    input answers
11    output “please enter the validation”
12    input validation
13    call validation function using the input as parameter validation (input)
14    storing returned valued from the validation function in a string variable
15    if “valid”
16    call random number generator function for number of question in a range 0 to 12
17   then according to the number call random question generator function
18   show the function
19   get the result from the used and store it in array
20   compare the give result with number (variable)
 23  end
24   show the final interface
```

End Function

Task-3

Now I include provide a few different kind of testing used for my provide pseudocode. I include given Black Box White Box and boundary testing.

Black Box testing for player name

Case	Details	Input	output
1	If player name less than 2 and greater 20	Ab	“Data name is valid”
2	If player name less than 2 and greater 20	Abc	“Data name is valid”
3	If player name less than 2 and greater 20	Abcd	“Data name is valid”
4	If player name less than 2 and greater 20	A	“Data name is invalid”

Black Box testing for question number

Case	Details	Input	Output
1	If question number less than 2 and greater 50	2	“Data name is valid”
2	If question number less than 2 and greater 50	4	“Data name is valid”
3	If question number less than 2 and greater 50	0	“Data name is invalid”
4	If question number less than 2 and greater 50	55	“Data name is invalid”

White box Testing for player name

Case	Input1	Input2	Expected	Actual
1	Ab	Ab	True	True
2	Abc	Abc	True	True
3	Abcd	Abcd	True	True
4	A	A	False	True

White box Testing for question number

Case	Input1	Input2	Expected	Actual
1	10	30	40	True
2	10	10	20	True

3	20	30	50	True
4	2	18	20	True
5	20	35	55	False
6	-50	20	-30	False
7	0	-15	-15	False
8	50	10	60	False

Boundary testing for player name

case	input	output	Actual result	Remarks
1	Ab	valid	Player name 2 less than and grater 20	correct
2	Abc	valid	Player name 2 less than and grater 20	correct
3	Abcd	valid	Player name 2 less than and grater 20	correct
4	A	invalid	Player name 2 less than and grater 20	correct

Boundary testing for question number

case	input	output	Actual result	Remarks
1	12	valid	Question number 2 less than and grater 50	Correct
2	23	Valid	Question number 2 less than and grater 50	Correct
3	20	Valid	Question number 2 less than and grater 50	Correct
4	55	Inalid	Question number 2 less than and grater 50	Correct
5	58	Invalid	Player name 2 less than and grater 20	Correct

Task-4

Desk checking for player name

Line number	Player name	Comment
1	Null	Function start
2	0	player name less than 2 and grater 20
3		Output “data player name is invalid”
4		End if
5		Else
6	3	Output “data player name is valid”
7		Return name
8		End function

Desk checking for question number

Line number	Question number	comment
1		Function start
2	4	if number less than 2 and grater 50
3	0	Output “data question number is invalid”
4		End if
5		else
6	4	Output “data question number is valid”
7	3	End function

Desk checking for question generator

Line number	First number	Last number	commend
1			Start function
2			Start loop
3	0	0	Data first number as whole number
4	0	0	Data last number as whole number
5	2	2	question less than 1and grater 7
6			Return true
7			End if
8			End loop
9			End function

Desk checking for addition

Line number	First Num	Last num	comment
1	0	0	Function start
2	0	0	Data result as whole number
3	0	0	Call numbergenerator
4	2	2	Result= firstnum+lastnum
5			Return result
6			End function

Desk checking for subtraction

Line number	First Num	Last num	comment
1	0	0	Function start
2	0	0	Data result as whole number

3	0	0	Call numbergenerator
4	4	4	Result=firstnum-lastnum
5	0	0	Return result
6			End function

Desk checking for division

Line number	First Num	Last num	commend
1	0	0	Function start
2	0	0	Data result as whole number
3	0	0	Call numbergenerator
4	2	2	Result=firstnum/lastnum
5	0	0	Return result
6			End function

Desk checking for multiplication

Line number	FirstNum	Last num	commend
1	0	0	Function start
2	0	0	Data result as whole number
3			Call numbergenerator
4	3	2	Result=firstnum*lastnum
5			Return result
6			End function

Desk checking for random generator

Line number	First number	Last number	comment
1			Function start
2	1	1	Data firstnum as whole number
3	1	1	Data lastnum as whole number

4	First=rand(0,12)
5	last=rand(0,12)
6	Return firstnum
7	Return lastnum
8	End function

Desk checking for question store

Line number	First Number	Last Number	Ans	Counter	Notas
1				0	Function start
2	0			0	Data first numberas whole number
3	0	0		0	Data last number as whole number
4	0	0	0	0	Start loop
5	0	0	0	1	first number less than 2 and grater or equal to 50 and
6	0	0	0	1	Last number less than 2 and grater or equal to 50
7					arrayfirst num[first number] less than arraylast num[last number]
8					arraylast num[last number] less than arrayfirst num [first num])
9					else
10					arrayfirst num[first number] summation of arraylast num[last

				number]
11	10	20	30	While loop
12	10	20	30	output
13				End if
14				End loop
15				End function

Desk checking for match result

Line number	First number	Last number	Result	counter	Comment
1					Start function
2	0	0	0	0	Data first number as whole number
3	0	0	0	0	Data last number as whole number
4	0	0	0	0	Result as whole number
5	0	0		0	Counter as whole number
6	2	2	2	2	Output fist number
7	3	3	3	3	Input first number
8					Output last number
9					Input last number
10					result=first number
11					Counter=1
12					Counter=last number
13					Result=result+first number
14					Next loop
15					End function

Desk checking for wrong answer

Line number	Wrong answer	commend
1		Start function
2	2	Data number less than 2 and 20
3	2	Return true
4		End if
5		else
6		Return false
7		Output result
8		End function

Desk checking for main function

Line number	Name	number	Validation	answer	commend
1	Null	Null	Null	Null	Data string
2	0	o	o	o	Data whole number
3					Output name
4					Input name
5	2	2	2	2	Output number
6	2	2	2	2	Input number
7	0	0	0	0	Answer whole number
8	0	0	0	0	Data validation whole number
9					Output answer

10		Input answer
11		Output validation
12		Input validation
13		User input
14		Return value
15		Call random number
16		call random question
17		Show the function
18		Used store array
19		Compare the give result
20		end
21		Show the final interface

Conclusion

In these task I have try to make up a successful project which I have use algorithms, test and desk check process to build it accurately. Although I have try to given best effort.

Reference:

1. http://www.answers.com/mt/function-generator
2. http://electronics.stackexchange.com/questions/tagged/function-generator
3. http://www.answers.com/mt/function-generator
4. http://stackoverflow.com/questions/17190340/find-a-random-number-generator-using-a-given-random-number-generating-function

5. Book reference
6. SDT student guide