**Total Marks: \_\_7.5\_\_\_\_\_\_\_\_\_**

**Obtained Marks: \_\_\_\_\_\_\_\_\_\_\_\_\_**

**Computer Organization & Assembly Language (Lab)**

**Project # 1**

**Quiz System**

**Submitted To: Engr. Muhammad Tauqeer Ali**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Group Members: Farooq Ahmed Khan (2012302)  
 Daniyal Shakeel (2012321)  
 Atika Khan (2012298)  
 Turab Ali Khan (2012339)  
 Fahad Wakeel (2012292)**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Contribution of Members**

|  |  |
| --- | --- |
| **Farooq Ahmed Khan** | **Variables and the code to use them** |
| **Daniyal Shakeel** | **Program Structure, Interrupts** |
| **Atika Khan** | **Registers** |
| **Turab Ali Khan** | **Operations (Calculations of marks)** |
| **Fahad Wakeel** | **Addressing Modes, keywords (other than**  **Basic ones)** |

**Quiz System Program in Assembly Language**

**Introduction:**

## This is a simple program in assembly language which is based on quiz system. Quiz system is a game concept. In this game, we have set 10 questions. For each right answer 1 mark will be increased and for each wrong answer 1 mark will be decreased from total points and at the end of the quiz, total marks will be shown.

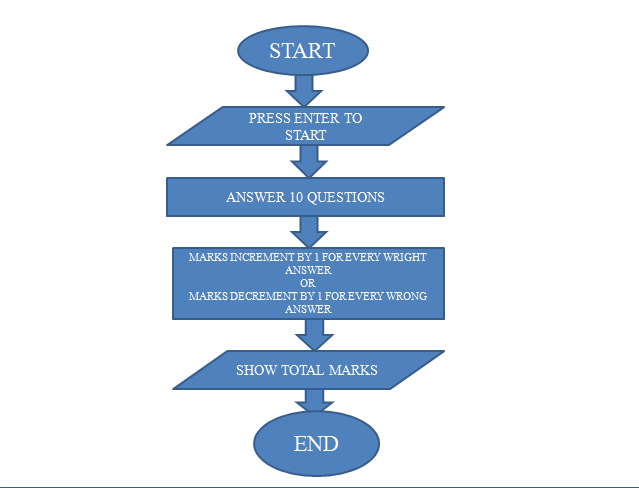
**Procedure:**

## Data Segment (Initialization and Definition of Variables): After the size allocation for program, and other necessary keywords, and in the .data section, we initialized variables. We have set 10 variables from msg 1 to msg9 for printing our desired strings and also, we have used Q1-Q9 and QA1-QA9 for printing our desired questions and their options.

## Code Segment (Choosing to enter into Quiz and Questions): Now we have taken a start level. At first, we have printed a new line and then we have taken an input to compare carriage return. If it is equal then it will go to the QSn1 level. Now we will go to the question 1 level and 1 input will be taken and it will be compared to the correct answer. If it is equal then it will go to the Qsn2 level otherwise it will reach to the Qsnw2 level. In QSN2 level, bl will be incremented for right answer then again, the same procedure for input will take place. In QSnw2 level bl will be decremented for wrong answer then again, the same procedure for input will take place.

* **Code Segment (Calculation Operations):**After all 10 submissions of MCQs, the program will calculate marks by incrementing and decrementing on every wrong and right answers by user. Then, the marks will be displayed. At the end, the program will display a message that will ask user if he wants to do another quiz or does he want to exit. If he replies with 1, the quiz will again start from num1 question, while if he replies with 0, the program will exit.

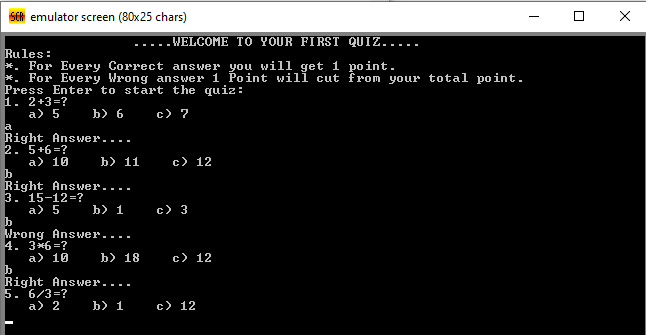
**Flow Chart:**

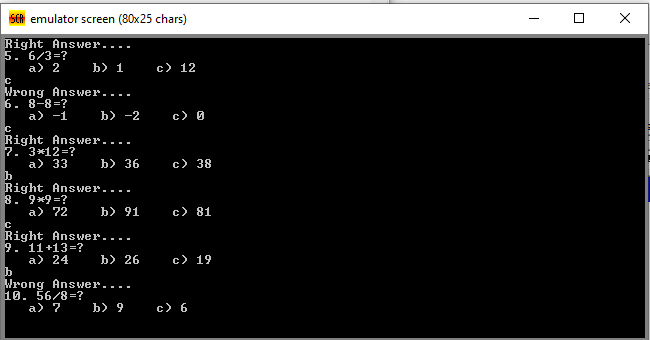


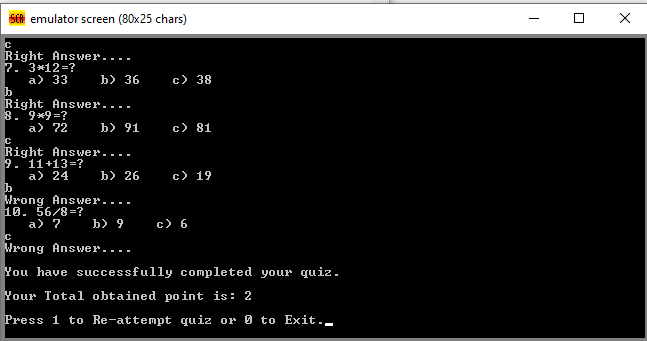
**Code:**

|  |
| --- |
| .MODEL SMALL  .STACK 100H  .DATA  MSG1 DB ' .....WELCOME TO YOUR FIRST QUIZ.....$'  MSG2 DB 'Rules : $'  MSG3 DB '\*. For Every Correct answer you will get 1 point.$'  MSG4 DB '\*. For Every Wrong answer 1 Point will cut from your total point.$'  MSG5 DB 'Press Enter to start the quiz : $'  MSG6 DB 'Right Answer....$'  MSG7 DB 'Wrong Answer....$'  MSG8 DB 'You have successfully completed your quiz.$'  MSG9 DB 'Your Total obtained point is : $'  MSG10 DB 'Press 1 to Re-attempt quiz or 0 to Exit.$'  MSG11 DB ' \*\*\*Thank you.! \*\*\*$'  Q1 DB '1. 2+3=?$'  QA1 DB ' a) 5 b) 6 c) 7$'  Q2 DB '2. 5+6=?$'  QA2 DB ' a) 10 b) 11 c) 12$'  Q3 DB '3. 15-12=?$'  QA3 DB ' a) 5 b) 1 c) 3$'  Q4 DB '4. 3\*6=?$'  QA4 DB ' a) 10 b) 18 c) 12$'  Q5 DB '5. 6/3=?$'  QA5 DB ' a) 2 b) 1 c) 12$'  Q6 DB '6. 8-8=?$'  QA6 DB ' a) -1 b) -2 c) 0$'  Q7 DB '7. 3\*12=?$'  QA7 DB ' a) 33 b) 36 c) 38$'  Q8 DB '8. 9\*9=?$'  QA8 DB ' a) 72 b) 91 c) 81$'  Q9 DB '9. 11+13=?$'  QA9 DB ' a) 24 b) 26 c) 19$'  Q10 DB '10. 56/8=?$'  QA10 DB ' a) 7 b) 9 c) 6$'  .CODE  MAIN PROC    MOV AX,@DATA  MOV DS,AX    LEA DX,MSG1  MOV AH,9  INT 21H    CALL NL    LEA DX,MSG2  MOV AH,9  INT 21H    CALL NL    LEA DX,MSG3  MOV AH,9  INT 21H    CALL NL    LEA DX,MSG4  MOV AH,9  INT 21H    START:  MOV BL, 0  CALL NL    LEA DX,MSG5  MOV AH,9  INT 21H      MOV AH, 1  INT 21H    CMP AL, 0DH  JE QSN1  JNE START    QSN1:  CALL NL    LEA DX,Q1  MOV AH,9  INT 21H    CALL NL    LEA DX,QA1  MOV AH,9  INT 21H    CALL NL    MOV AH, 1  INT 21H  CMP AL, 'a'  JE QSN2  JNE QSNW2    QSN2:  CALL NL    LEA DX,MSG6  MOV AH,9  INT 21H    INC BL  CALL NL    CALL QN2    CALL INPUT    CMP AL, 'b'  JE QSN3  JNE QSNW3    QSNW2:  CALL NL    LEA DX,MSG7  MOV AH,9  INT 21H    DEC BL  CALL NL    CALL QN2  CALL INPUT    CMP AL, 'b'  JE QSN3  JNE QSNW3      QSN3:  CALL NL    LEA DX,MSG6  MOV AH,9  INT 21H    INC BL  CALL NL    CALL QN3  CALL INPUT    CMP AL, 'c'  JE QSN4  JNE QSNW4    QSNW3:  CALL NL    LEA DX,MSG7  MOV AH,9  INT 21H    DEC BL  CALL NL    CALL QN3  CALL INPUT    CMP AL, 'c'  JE QSN4  JNE QSNW4    QSN4:  CALL NL    LEA DX,MSG6  MOV AH,9  INT 21H    INC BL  CALL NL    CALL QN4  CALL INPUT    CMP AL, 'b'  JE QSN5  JNE QSNW5    QSNW4:  CALL NL    LEA DX,MSG7  MOV AH,9  INT 21H    DEC BL  CALL NL    CALL QN4  CALL INPUT    CMP AL, 'b'  JE QSN5  JNE QSNW5    QSN5:  CALL NL    LEA DX,MSG6  MOV AH,9  INT 21H    INC BL  CALL NL    CALL QN5    CALL INPUT    CMP AL, 'a'  JE QSN6  JNE QSNW6    QSNW5:  CALL NL    LEA DX,MSG7  MOV AH,9  INT 21H    DEC BL  CALL NL    CALL QN5  CALL INPUT    CMP AL, 'a'  JE QSN6  JNE QSNW6    QSN6:  CALL NL    LEA DX,MSG6  MOV AH,9  INT 21H    INC BL  CALL NL    CALL QN6    CALL INPUT    CMP AL, 'c'  JE QSN7  JNE QSNW7    QSNW6:  CALL NL    LEA DX,MSG7  MOV AH,9  INT 21H    DEC BL  CALL NL    CALL QN6  CALL INPUT    CMP AL, 'c'  JE QSN7  JNE QSNW7    QSN7:  CALL NL    LEA DX,MSG6  MOV AH,9  INT 21H    INC BL  CALL NL    CALL QN7  CALL INPUT    CMP AL, 'b'  JE QSN8  JNE QSNW8    QSNW7:  CALL NL    LEA DX,MSG7  MOV AH,9  INT 21H    DEC BL  CALL NL    CALL QN7  CALL INPUT    CMP AL, 'b'  JE QSN8  JNE QSNW8    QSN8:  CALL NL    LEA DX,MSG6  MOV AH,9  INT 21H    INC BL  CALL NL    CALL QN8  CALL INPUT    CMP AL, 'c'  JE QSN9  JNE QSNW9    QSNW8:  CALL NL    LEA DX,MSG7  MOV AH,9  INT 21H    DEC BL  CALL NL    CALL QN8  CALL INPUT    CMP AL, 'c'  JE QSN9  JNE QSNW9    QSN9:  CALL NL    LEA DX,MSG6  MOV AH,9  INT 21H    INC BL  CALL NL    CALL QN9  CALL INPUT    CMP AL, 'a'  JE QSN10  JNE QSNW10    QSNW9:  CALL NL    LEA DX,MSG7  MOV AH,9  INT 21H    DEC BL  CALL NL    CALL QN9  CALL INPUT    CMP AL, 'a'  JE QSN10  JNE QSNW10    QSN10:  CALL NL    LEA DX,MSG6  MOV AH,9  INT 21H    INC BL  CALL NL    CALL QN10  CALL INPUT    CMP AL, 'a'  JE EXIT  JNE EXITW    QSNW10:  CALL NL    LEA DX,MSG7  MOV AH,9  INT 21H    DEC BL  CALL NL    CALL QN10  CALL INPUT    CMP AL, 'a'  JE EXIT  JNE EXITW    EXIT:  CALL NL    LEA DX,MSG6  MOV AH,9  INT 21H    INC BL  CALL NL  CALL NL    LEA DX,MSG8  MOV AH,9  INT 21H    CALL NL    LEA DX,MSG9  MOV AH,9  INT 21H    ADD BL, 48    CMP BL,57  JG TEN  MOV AH, 2  MOV DL, BL  INT 21H  JMP EXIT1    EXITW:  CALL NL    LEA DX,MSG7  MOV AH,9  INT 21H    DEC BL  CALL NL  CALL NL    LEA DX,MSG8  MOV AH,9  INT 21H    CALL NL  CALL NL    LEA DX,MSG9  MOV AH,9  INT 21H    ADD BL,48  MOV AH,2  MOV DL, BL  INT 21H    JMP EXIT1    TEN:  MOV AH,2  MOV DL,"1"  INT 21H  MOV DL,"0"  INT 21H  JMP EXIT1    NL:  MOV AH,2  MOV DL, 0AH  INT 21H  MOV DL, 0DH  INT 21H  RET    QN2:  LEA DX,Q2  MOV AH,9  INT 21H    CALL NL    LEA DX,QA2  MOV AH,9  INT 21H  RET    QN3:  LEA DX,Q3  MOV AH,9  INT 21H    CALL NL    LEA DX,QA3  MOV AH,9  INT 21H  RET    QN4:  LEA DX,Q4  MOV AH,9  INT 21H    CALL NL    LEA DX,QA4  MOV AH,9  INT 21H  RET    QN5:  LEA DX,Q5  MOV AH,9  INT 21H    CALL NL    LEA DX,QA5  MOV AH,9  INT 21H  RET    QN6:  LEA DX,Q6  MOV AH,9  INT 21H    CALL NL    LEA DX,QA6  MOV AH,9  INT 21H  RET    QN7:  LEA DX,Q7  MOV AH,9  INT 21H    CALL NL    LEA DX,QA7  MOV AH,9  INT 21H  RET    QN8:  LEA DX,Q8  MOV AH,9  INT 21H    CALL NL    LEA DX,QA8  MOV AH,9  INT 21H  RET    QN9:  LEA DX,Q9  MOV AH,9  INT 21H    CALL NL    LEA DX,QA9  MOV AH,9  INT 21H  RET    QN10:  LEA DX,Q10  MOV AH,9  INT 21H    CALL NL    LEA DX,QA10  MOV AH,9  INT 21H  RET    INPUT:  CALL NL    MOV AH, 1  INT 21H  RET      EXIT1:  CALL NL  CALL NL    LEA DX,MSG10  MOV AH,9  INT 21H    MOV AH,1  INT 21H    CMP AL,'1'  JE START    CALL NL  CALL NL    LEA DX,MSG11  MOV AH,9  INT 21H    MOV AH, 4CH  INT 21H    MAIN ENDP  END MAIN |

**Screen Shots:**

****

****

****

**Problems:**

We are facing a problem, that is: the program is sometimes returning right marks and sometimes wrong. We tried but could not reach the root of the problem, and hence it remains a headache.

**Conclusions:**

The program did run smoothly after an exhausting struggle. We have found keywords very confusing and hard to remember as they’re not based on straightforward English like high level languages. Also, the concepts around registers, addressing modes, ASCII codes are confusing too, and it’s hard to grasp one’s mind around it.

We conclude that people who made high level languages are the very people because of whom the world is better as we experience today, and we also share our sympathy with those who had to work with assembly language.  
Humor apart, we have learned much more about Assembly language and its keywords, components etc. by practicing it in this project.