KIET Group of Institutions, Ghaziabad

COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

PROJECT BASED LEARNING

On

QUIZ GAMES

SUBJECT: DATA STRUCTURE AND ALGORITHM LAB

(KCS – 351)

Submitted By:
ISHAN SHARMA – 2100290110065 (CSIT 3A)
ASTHA GUPTA – 2100290110038 (CSIT 3A)
ANUSHKA GUPTA – 2100290110033 (CSIT 3A)

ACKNOWLEDGEMENT

I've got this golden opportunity to express my kind gratitude and sincere thanks to my subject faculty "**Mr. Vinay Kumar**", Computer Science and Information Technology Department, **KIET GROUP OF INSTITUTIONS** for their kind support and necessary counselling in the preparation of this project report. I'm also indebted to each and every personresponsible for the making up of this project directly or indirectly.

I must also acknowledge or deep debt of gratitude each one of my colleague who led this project come out in the way it is. It's my hard work and untiring sincere efforts and mutual cooperation to bring out the project work. Last but not the least, I would like to thank my parents for their sound counselling and cheerful support. They have always inspired us and kept our spirit up.

INDEX

AIM

Quiz is a form of game or mind sport in which players attempt to answer questions correctly on one or several specific topics. Quizzes can be used as a brief assessment in education and similar fields to measure growth in knowledge, abilities, or skills.

OBJECTIVE

The Objective behind to organize Quiz game is to evaluate the knowledge of the participants within academics as well as beyond academics and to make them familiar with the prospects of quizzes and the objectivity of the questions. The main purpose of the game to develop interest in subject areas of Commerce and Management including competitive aspects.

ABSTRACT

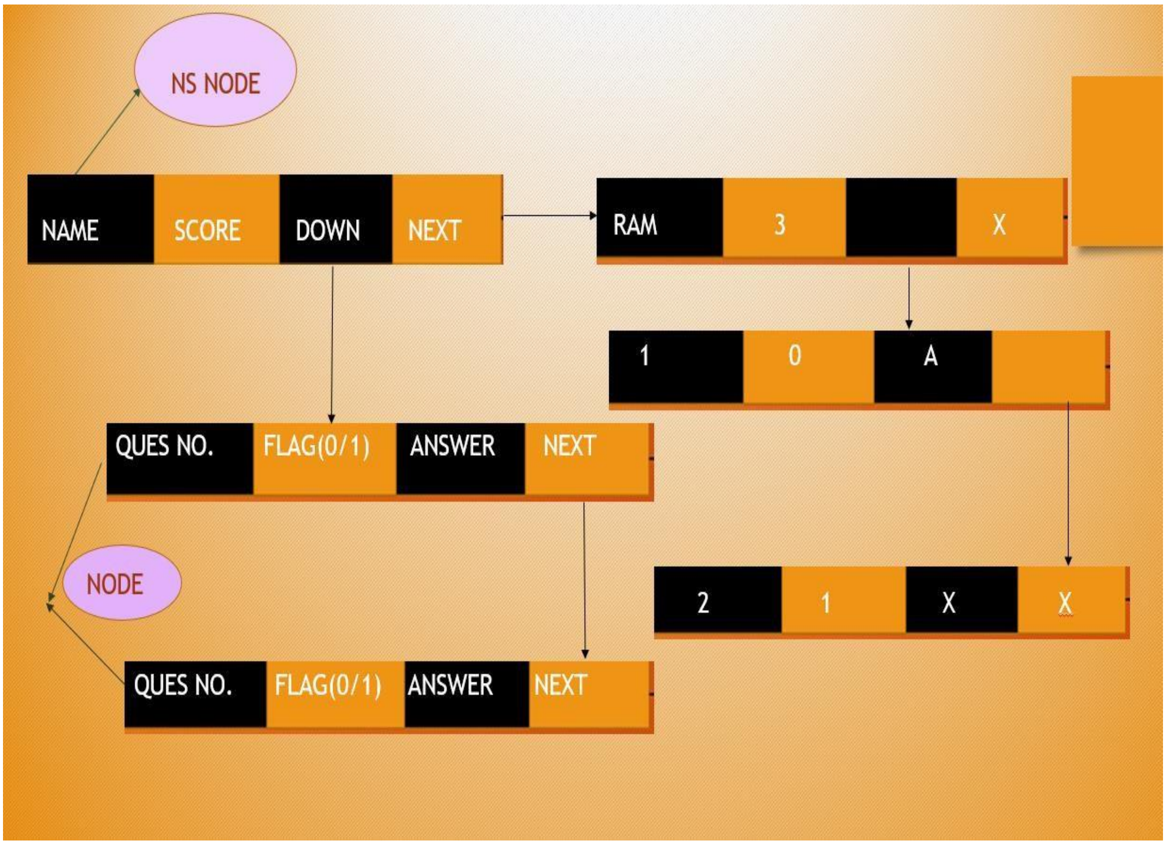
The concept of quizzes is currently very popular among educated circles as well as in entertainment shows.

Though the quiz can be conducted manually, it often needs elaborate preparations. Quizzes contribute to the growth of knowledge of an individual and they are a popular source of entertainment. This program In C++ focuses on creating interactive quizzes possibly with a large database of questions from various subjects. The program utilizes most of the important concepts of Data Structures . The final output is envisioned to be a user- friendly interactive quiz with which the user can gain significant knowledge and get entertainment.

BASIC PRINCIPLES:

- **LINKEDLIST** - a linked list consists of nodeswhere each node contains a data field and a reference(link) to the next node in the list.
- **VECTOR - Vectors** in **C++** are sequence containers representing arrays that
- can change theirsize during runtime
- **SORTING -** Sorting is the process of arranging elements either in ascending
- descending order

DIAGRAMATIC REPRESENTATIOIN OF LINKEDLIST:



CODING IMPLEMENTATION

```
#include<iostream>
#include<vector>
#include<bits/stdc++.h>   using
namespace std;

struct node{

            char   corr_ans;  int
            ques;
            int flag;

            struct node *next;

};


struct NS_node{                                    //NS node : Name ScoreNode

            char
            *name;int
            score;
            struct NS_node *next;struct
      node *down;
}*start;
```

```
struct NS_node* head = NULL;



void insert(char* str) {

    struct NS_node* newNode = (struct NS_node*) malloc(sizeof(struct
NS_node));

    newNode->name = (char*) malloc(strlen(str) + 1); strcpy(newNode->name,
    str);
    newNode->next = head; newNode-
    >score = 0; newNode->down =
    NULL;head = newNode;
}

bool sortbysec(const pair<string,int> &a,constpair<string,int> &b)

{

    return (a.second > b.second);

}

int main(){

        int players, temp, ctr=0, pcount=1, pcounter=0, posi=1;char names[30], a1, a2, a3, a4,
        a5;
        cout<<"\t\t\t\t______________________________________________\n";
        cout<<"\t\t\t\tDATA STRUCTURE QUIZ GAME\n";cout<<"\t\t\t\t\n";
        cout<<"Hey! Welcome to DS quiz game.\nEnter thenumber of players: ";

        cin>>players;
        temp = players;
        cout<<"Enter names of all players: \n\n";while(players>0){
                cout<<"Enter Player "<<pcount<<":";cin>>names;
                insert(names
                );   pcount++;
                players--;
        }

        struct NS_node* ptr;ptr=
    head;
        while(ptr!=NULL){                                       //check



                cout<<"\n\nLets Begin "<< ptr-
>name<<"!!"<<endl<<endl ;

                cout<< "Q1.   A procedure that calls itself is called:
\nA. Illegal Call\nB. Reverse Polish\nC. Recursive\nD.None\nAnswer:" ;

                cin>>a1;
                cout<<"\
                n";
                struct node* ques1;

                ques1 = (struct node *)malloc(sizeof(struct node));
```

```
ques1->ques = 1;

ques1->corr_ans  =  NULL; ques1-
>next = NULL;
ptr->down = ques1;


if   (a1=='c'   ||   a1=='C'){
        ques1->flag  =  1;
        ctr++;


}

else{

        ques1->flag   =   0;   ques1-
        >corr_ans = 'C';
}


cout<< "Q2. What data structure is used for depth first traversal of a graph?:
\nA. Queue\nB. Stack\nC. List\nD.None\nAnswer:" ;

cin>>a2;
cout<<en
dl;
struct node* ques2;

ques2 = (struct node *)malloc(sizeof(struct node));ques2->next = NULL;
ques1->next    =    ques2;
ques2->ques = 2;
```

```
        ques2->corr_ans     =     NULL;    if
        (a2=='b' || a2=='B'){
                ques2->flag  =  1;
                ctr++;


        }

        else{

                ques2->flag   =   0;   ques2-
                >corr_ans = 'B';
        }

        cout<< "Q3. Which one of the below is not divideand conquer approach?: \nA.
Insertion Sort\nB. Merge Sort\nC. Shell Sort\nD. Heap Sort\nAnswer:" ;

        cin>>a3;
        cout<<en
        dl;
        struct node* ques3;

        ques3 = (struct node *)malloc(sizeof(struct node));ques3->next = NULL;
        ques2->next     =     ques3;
        ques3->ques = 3;
        ques3->corr_ans     =     NULL;    if
        (a3=='b' || a3=='B'){
                ques3->flag  =  1;
                ctr++;
        }
```

```
else{

        ques3->flag  =  0;  ques3-
        >corr_ans = 'B';
}

cout<< "Q4. Program with highest run-time complexity is: \nA. Tower of
Hanoi\nB. Fibonacci Series\nC.Prime Number Series\nD. None of the above\nAnswer:" ;

cin>>a4;
cout<<en
dl;
struct node* ques4;

ques4 = (struct node *)malloc(sizeof(struct node));ques4->next = NULL;
ques3->next   =   ques4;
ques4->ques = 4;
ques4->corr_ans   =   NULL;   if
(a4=='a' || a4=='A'){
        ques4->flag = 1;
        ctr++;


}

else{

        ques4->flag  =  0;  ques4-
        >corr_ans = 'A';
}
```

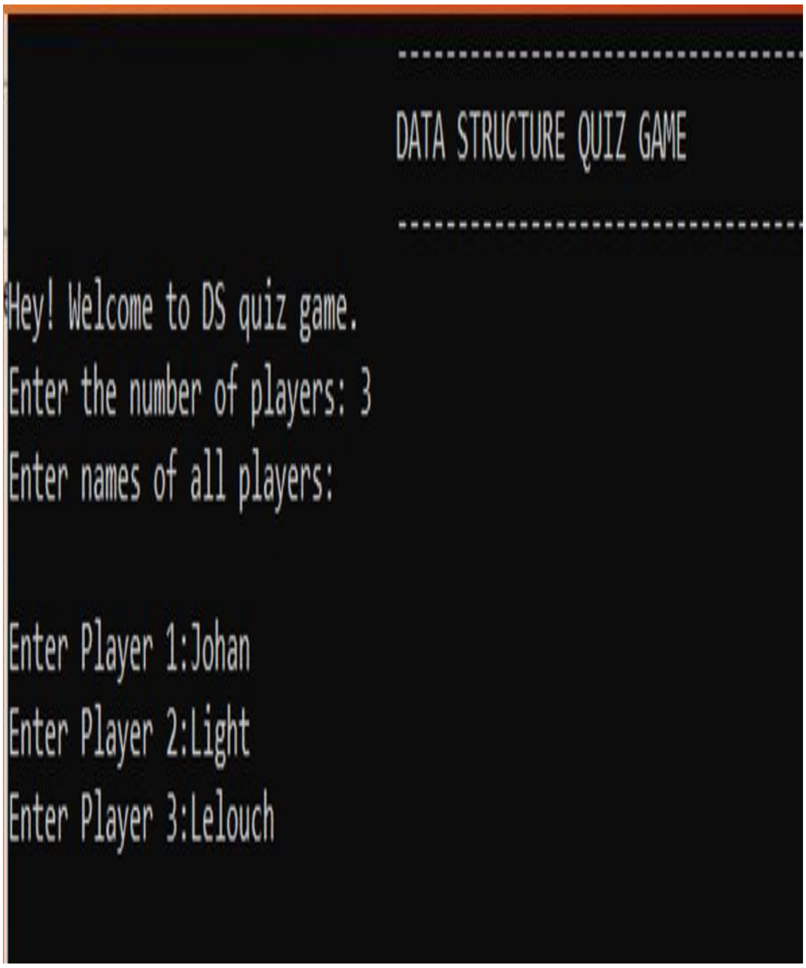
```
                    cout<< "Q5. Stack is used for: \nA. CPU Resource Allocation\nB. Breadth First
Traversal\nC. Recursion\nD. None of the above\nAnswer:" ;

                    cin>>a5;
                    cout<<en
                    dl;
                    struct node* ques5;

                    cout<<"__________________________________________________________________
______________"<<endl;

                    ques5 = (struct node *)malloc(sizeof(struct node));ques5->next = NULL;
                    ques4->next    =    ques5;
                    ques5->ques = 5;
                    ques5->corr_ans    =    NULL;    if
                    (a5=='c' || a5=='C'){
                              ques5->flag  =  1;
                              ctr++;


                    }

                    else{

                              ques5->flag   =   0;   ques5-
                              >corr_ans = 'C';
                    }

                    ptr->score  =  ctr;
                    ctr = 0;
                    ptr=ptr->next;
```

```
}

struct  NS_node*  save  =  head; cout<<"
 ---------------------------- "<<endl;
cout<<"RESULTS:"<<endl;while(save!=NULL){

    cout<<" ------------------"<<endl;

        cout<<"\nName:   "<<save->name<<"\n";   cout<<"Score:   "<<save-
        >score<<"\n"<<endl;if (save->score != 5){
            struct          node          *movedown;
            movedown        =        save->down;
            cout<<"Corrections: ";
        while(movedown!=NULL){
                if(movedown->flag == 0)
                    {

                cout<<"\nQuestion      "<<movedown->ques<<":";     cout<<movedown-
                    >corr_ans;
                    }



                movedown=movedown->next;

            }

    }
    save=save->next;
    cout<<endl;
}

//getting all the names
ptr=head;
string namez[1000]; for(int
i=0;i<temp;i++){
    namez[i]=ptr->name;
    ptr=ptr->next;
}


ptr=head;

int finalScore[temp]; for(int
i=0;i<temp;i++){
    finalScore[i]=ptr->score; ptr=ptr-
    >next;
}

vector< pair <string, int> > vect; for (int i=0;
i<temp; i++)
    vect.push_back( make_pair(namez[i],finalScore[i]) );


sort(vect.begin(), vect.end(), sortbysec);cout<<endl;
cout << "\nLEADERBOARD: \n"<<endl ;
```

```
	for (int i=0; i<temp; i++){

		cout <<posi<<". "<< vect[i].first << " "<< vect[i].second
<< endl;

		posi++;

	}

	return 0;



}
```

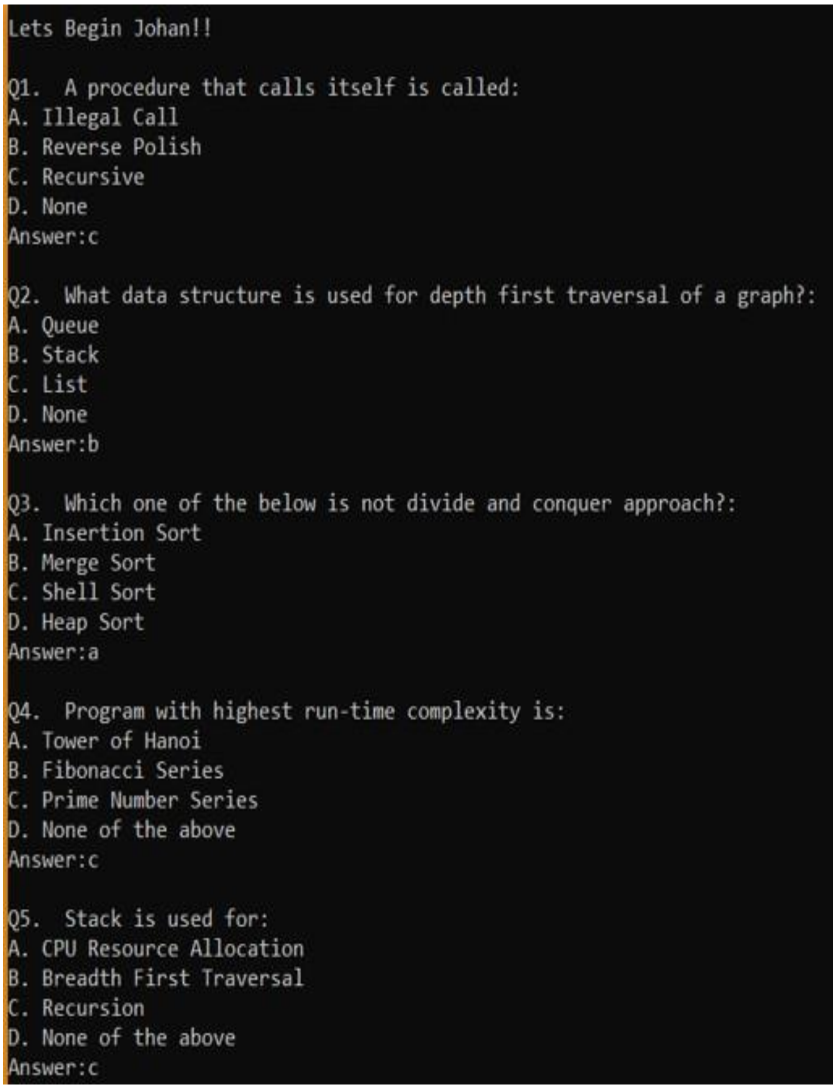
OUTPUT SCREENSHOT

Entering name:

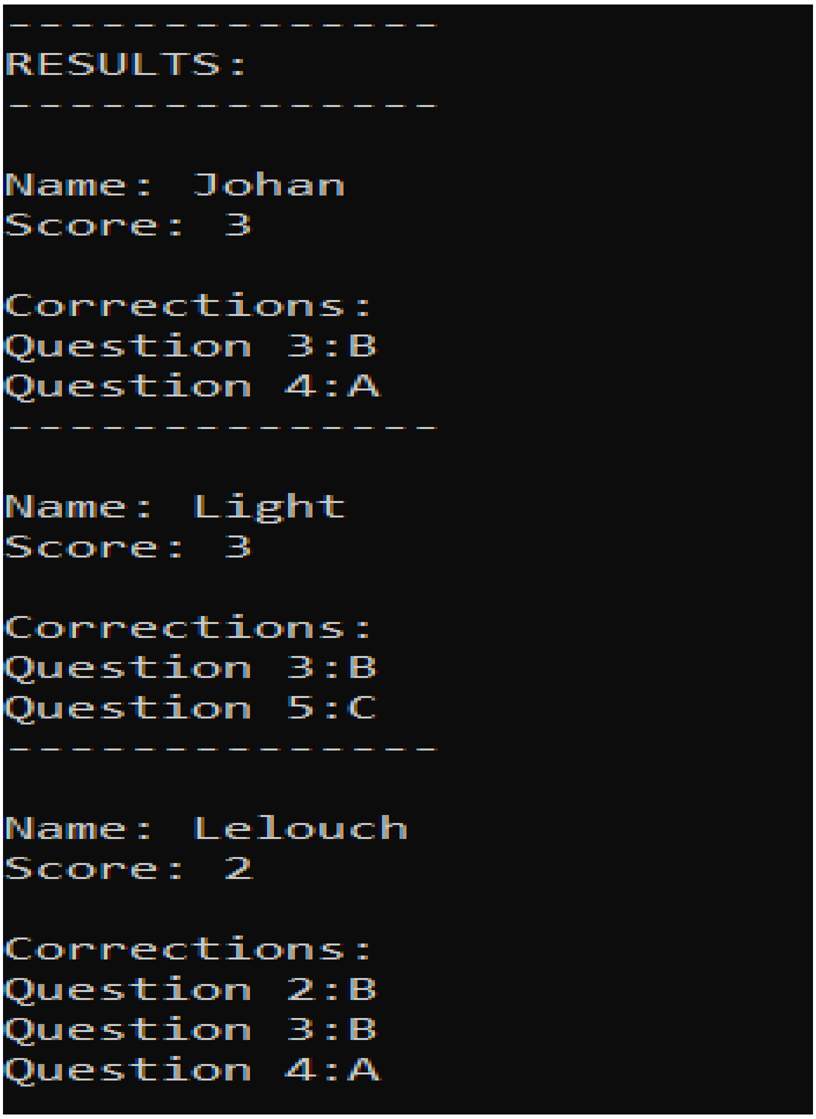


```
                                ----------------------------------
                                DATA STRUCTURE QUIZ GAME
                                ----------------------------------
Hey! Welcome to DS quiz game.
Enter the number of players: 3
Enter names of all players:

Enter Player 1:Johan
Enter Player 2:Light
Enter Player 3:Lelouch
```

QUESTIONS:

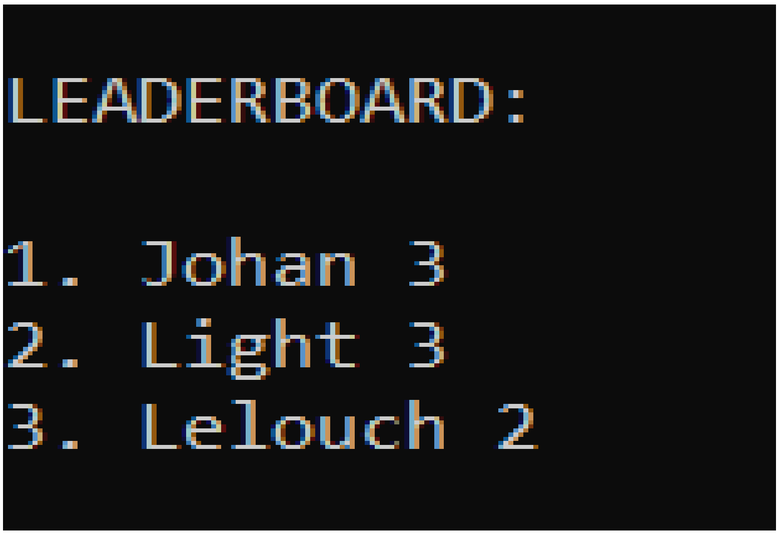


```
Lets Begin Johan!!

Q1.  A procedure that calls itself is called:
A. Illegal Call
B. Reverse Polish
C. Recursive
D. None
Answer:c

Q2.  What data structure is used for depth first traversal of a graph?:
A. Queue
B. Stack
C. List
D. None
Answer:b

Q3.  Which one of the below is not divide and conquer approach?:
A. Insertion Sort
B. Merge Sort
C. Shell Sort
D. Heap Sort
Answer:a

Q4.  Program with highest run-time complexity is:
A. Tower of Hanoi
B. Fibonacci Series
C. Prime Number Series
D. None of the above
Answer:c

Q5.  Stack is used for:
A. CPU Resource Allocation
B. Breadth First Traversal
C. Recursion
D. None of the above
Answer:c
```



```
---------------
RESULTS:
---------------

Name: Johan
Score: 3

Corrections:
Question 3:B
Question 4:A
---------------

Name: Light
Score: 3

Corrections:
Question 3:B
Question 5:C
---------------

Name: Lelouch
Score: 2

Corrections:
Question 2:B
Question 3:B
Question 4:A
```

The Result Card of eachstudent is showed, along with the correctionto their wrong answer.



```
LEADERBOARD:

1. Johan 3
2. Light 3
3. Lelouch 2
```

Names and scores are structured in a Vector Pair, andthen are sorted to display the Leaderboard**.**

TIME COMPLEXITY: O (NlogN)

SPACE COMPLEXITY: O (N)

GitHub Link:

https://github.com/IshanSharma137/QuizGameDS/blob/main/QuizGameDS.cpp

REFERENCES:

1. https://www.geeksforgeeks.org/sort-vector-of-pairs-in-ascending-order-in-c/
2. https://www.youtube.com/watch?v=FY1LEP