Department of Computer Engineering

Programming for Engineers

Fall 21-22

TERM PROJECT

Group Members

- **Kemalettin Koçyiğit, 64180003, CoE**
- **Gülsüm İkbal Avşar, 64180006,CoE**
- **Mehmet Emre Ayar, 64180009, CoE**

1. Summary of Project

The main theme of project is designing a *BookStore* program which mainly keeps track of list of some books and customers. In the list of books, there are some information about book such as book name, price of it and current stock status. According to these informations, some methods are operated. These methods are *printBooks*, *checkBook*, *sellBook*, *addBook* etc.

```
                                =
                               ======
                              ============     _____
                            ==================  |   |
                          ======================|   |
                        ==========================|   |
                      ============================|   |==
                    ========================================
                  ==============================================
          ==================  My Book Store  ===================
=============================================================
||   Kurk Mantolu Madonna     ||     $19.90     ||   2    ||
||   Son Ada                  ||     $9.90      ||   3    ||
||   Nutuk                    ||     $49.90     ||   20   ||
||   Suc ve Ceza              ||     $9.90      ||   40   ||
||   Gurur ve Onyargi         ||     $29.90     ||   30   ||
||   Ermis                    ||     $9.90      ||   30   ||
=============================================================
```

printBooks()'s output

```
        /\
        /
       / Queue Start
      /
     /
    1       Mehmet Emre Ayar
      0
     /|\
     /.\
    2       Gulsum Ikbal Avsar
      0
     /|\
     /.\
    3       Kemalettin Kocyigit
      0
     /|\
     /.\
    4       Roberto Carlos
      0
     /|\
     /.\
    5       Lebron James
      0
     /|\
```

When user wants to display all the books in the list, *printBooks* method is included. It displays book name, price of it and number of books in the stock. Moreover, before the selling book and adding book to the list, *checkBook* method becomes involved. Before selling book, *checkBook* method controls whether the book is in the list. Also, if there is book which wanted to be sold in the list, *checkBook* controls the stock of it. Thus, if there is no stock, there should not be any selling. In the addBook method, checkBook is used for controlling the list whether the book which is wanted to be added is already in the list. If book is already there, stock of it is increasing. If there is no such that book, the book is added to the list.

```
vector<vector<string>> books{{"Kurk Mantolu Madonna","$19.90","2"},{"Son Ada","$9.90","3"},{"Nutuk","$49.90","20"},
                             {"Suc ve Ceza","$9.90","40"},{"Gurur ve Onyargi","$29.90","30"},{"Ermis","$9.90","30"}};
```

Books are stored in a vector which is 2D. The reason of using vectors instead of arrays is that vectors are acts like dynamic arrays. It means that vectors have the ability to resize itself automatically if an element is added or deleted. In arrays, adding or deleting is a long process because array size is constant at the beginning. To insert an array, the new array with inserted size should be created, and every element should be placed to this new array one by one. In the vectors, it is easier. While adding an element to vectors, new element is inserted to the end.

While 2-D vector is created in main function, the type of data is determined as *string* because all informations are written as *string*. Then, the name of vector is written as *book*. In the vector, name of book, price and stock value are written for each book.

```
void printBooks(vector<vector<string>> vec){

    for (int i = 0; i < vec.size(); i++)
    {
        for (int j = 0; j < vec[i].size(); j++)
        {
            cout << vec[i][j] << " ";
        }
        cout << endl;
    }
}
```

```
Kurk Mantolu Madonna  $19.90  2
Son Ada  $9.90  3
Nutuk  $49.90  20
Suc ve Ceza  $9.90  40
Gurur ve Onyargi  $29.90  30
Ermis  $9.90  30
```

To display all books and information about that book, *printBooks* method is used. The vector that we created in main function is the only parameter of this method. Since there is nothing to return, method is written as *void*. The nested for-loop is used because there is more than one dimension. In the first for loop, the rows of vector are looping. Thus, *i* is the index of row which is from 0 until one less of row number. In the inner for-loop, the columns are looping. For example, when the loops are starting to work, row 0 column 0, row 0 column 1, row 0 column 2, …., row 2 column 2 etc. will display respectively.

```
bool checkBook(string name, vector<vector<string>> vec, int t = 0) {
    for (int i = 0; i < vec.size(); i++) {

        if (name == vec[i][0]) {
            cout << name << " is available..." << endl;
            t = 1;
            return true;
        }
    }
    if (t == 0) {
        cout << name << " is not available...";
        return false;
    }
    else{
        return false;
    }
}
```

```
checkBook( name: "Suc ve Ceza", vec: books);

Suc ve Ceza is available...

checkBook( name: "Seker Portakali", vec: books);

Seker Portakali is not available...
```

The *checkBook* method has 3 parameters which are book name, vector of book and an integer t to indicate the output. There is only one for-loop, and it is used to get the rows. In *checkBook* method, the name of book is important, and we know that the index of book names is in column 0. So, it is only required to check the column 0 for each row. If the name is matched with the name which is parameter for method, the output displays *"…. is available..",* t value becomes 1 and method returns true. In the case of names are not matched, t value stays 0 and if condition become involved. Output is *"… is not available…"*, and method returns false. The importance of returning Boolean is for other methods. In the beginning, we mentioned that *checkBook* method will be used as condition for other methods. If this method returns true, it means that there is a book that user wants to buy or add to the list.

```
vector<vector<string>> sellBook(string name, vector<vector<string>> vec,
                                int amount){

    bool isAvailable = checkBook( name: name, vec: vec);

    if (isAvailable){

        for(int i = 0; i<vec.size(); i++) {

            if (vec[i][0] == name){
                int c = ((atoi( _Str: vec[i][2].c_str())) - amount);
                stringstream ss;
                ss << c;
                string s;
                ss >> s;
                if (c == 0){
                    cout << "The last book has been sold..." << endl;
                    vec.erase(vec.begin() + i);
                }else if (c < 0){
                    cout << "The book that trying to be bought cannot be supplied."
                            " There is not enough book to sale." << endl;
                }else{
                    vec[i][2] = s;
                }

            }
        }
    }else{
        cout << "The book you were looking for was not found.";
    }

    return vec;
}
```

For the selling book, *sellBook* method is used. It should be returned vector because we will update the vector of books. As parameters, string *name* of book, the vector of books and integer *amount* of selling are taken from user. First thing to do is controlling the book whether there is a such that book or we have stock in the bookstore. *checkBook* method is involved there, and the Boolean result of this method is assigned to a variable called *isAvailable*. After then, if-else condition blocks are become involved. If it is true, the name of book is being searched using for-loop. When names are matched, the stock value of book which is in the last column in the vector (column 2) is converted to int from string. Then, stock is decreased about *amount*. If the result is not less than 0, the selling is not allowed and output will be *"The book that trying to be bought cannot be supplied. There is not enough book to sale."* or if the result is "0", then the selling is allowed and the book that is sold , is removed from the vector and output will be *"The last book has been sold..."*.Lastly, if the result is more than "0", the result is assigned to an integer variable. After all, this integer result is converted to string again, and placed to vector. On the other hand, if there is no such a book in the list, output will be *"The book you were looking for was not found."*.

```
books = sellBook( name: "Gurur ve Onyargi", vec: books, amount: 3);
printBooks( vec: books);
```

```
Gurur ve Onyargi is available...
Kurk Mantolu Madonna  $19.90  2
Son Ada  $9.90  3
Nutuk  $49.90  20
Suc ve Ceza  $9.90  40
Gurur ve Onyargi  $29.90  27
Ermis  $9.90  30
```

The *addBook* method is providing that if the book wanting to add is not in the bookstore, book is added to the bookstore with its price and stock amount. If book is already in the bookstore, the stock amount is increased as *amount*. Thus, the parameters of *addBook* are *name* of book as string, *price* as string, stock *amount* as integer and the book vector defined in main function. Again, the first thing to do is checking book whether the book is already in the list or not. If it is in the bookstore, only thing to do is to increase the stock of book about amount. This process is the same with *sellBook* methods. However, if we do not have this book in the store, it should be added to the vector using *push_back()* method of vectors. *Name* of book, *price* and stock *amount* are assigned to this *push_back()* method. The important point is that at the beginning of *addBook* method, the stock *amount* is integer. While *push_back()* method, it should be converted to string because we defined the vector elements as string at the main function. When the new book is pushed back to the vector, it is added to the end of the vector. Finally, the updated vector is returned.

```
vector<vector<string>> addBook(string name,string price, vector<vector<string>> vec, int amount){

    bool isAvailable = checkBook( name: name, vec: vec);

    if (isAvailable){
        for(int i = 0; i<vec.size(); i++) {
            if (vec[i][0] == name) {
                int c = ((atoi( _Str: vec[i][2].c_str())) + amount);
                stringstream ss;
                ss << c;
                string s;
                ss >> s;
                vec[i][2] = s;
            }
        }

        return vec;
    }
    else{

        vec.push_back({name,price, to_string(amount)});


        return vec;
    }
}
```

```
books = addBook( name: "Seker Portakali", price: "$12.90", vec: books, amount: 5);
printBooks( vec: books);
```

```
Seker Portakali is not available...
Kurk Mantolu Madonna  $19.90  2
Son Ada  $9.90  3
Nutuk  $49.90  20
Suc ve Ceza  $9.90  40
Gurur ve Onyargi  $29.90  30
Ermis  $9.90  30
Seker Portakali  $12.90  5
```

```
books = addBook( name: "Nutuk", price: "$12.90", vec: books, amount: 10);
printBooks( vec: books);
```

```
Nutuk is available...

Kurk Mantolu Madonna  $19.90  2
Son Ada  $9.90  3
Nutuk  $49.90  30
Suc ve Ceza  $9.90  40
Gurur ve Onyargi  $29.90  30
Ermis  $9.90  30
```

```
queue<string> line;
line.push( x: "Mehmet Emre Ayar");
line.push( x: "Gulsum Ikbal Avsar");
line.push( x: "Kemalettin Kocyigit");
line.push( x: "Roberto Carlos");
line.push( x: "Lebron James");
line.push( x: "Rene Magritte");
line.push( x: "Jason Statham");
line.push( x: "Christopher Nolan");
line.push( x: "Stanley Kubrick");
line.push( x: "David Fincher");
line.push( x: "Ted Mosby");
line.push( x: "Chandler Bing");
```

To observe real life story, the bookstore has a queue structure to serve one by one to customer. The reason using queue is that the queue structure has a rule that the data only can be extracted first index

or last index. In that case, first input data will be first output in this situation. This operation is called FIFO (First In First Out).

```
void printline(queue<string> sira,int t = 1)
{

    queue<string> g = sira;
    while (!g.empty()) {

        cout << to_string(t) << '\t' << g.front() << endl;
        cout << "  O" << endl << " /|\\" << endl << " /.\\" << endl;
        g.pop();
        t++;
    }
    cout << '\n';
}
```

To print the line of the bookstore the function called “printline()” is generated.

```
cout << "  O" << endl << " /|\\" << endl << " /.\\" << endl;
```

This code line provide to draw a basic stickman to visualize the customers.

The Story of Bookstore

According to the minimum requirement of the project, the story scheme has generated down below by using the output.

```
                                =
                               ======
                              ============     _____
                             ==================  |   |
                            =======================|   |
                           ========================|   |
                          ============================|   |==
                        ==========================================
                      ================================================
                   ==================  My Book Store  ===================
============================================================
||   Kurk Mantolu Madonna     ||     $19.90     ||   2    ||
||   Son Ada                  ||     $9.90      ||   3    ||
||   Nutuk                    ||     $49.90     ||   20   ||
||   Suc ve Ceza              ||     $9.90      ||   40   ||
||   Gurur ve Onyargi         ||     $29.90     ||   30   ||
||   Ermis                    ||     $9.90      ||   30   ||
============================================================
     /\
     /
    / Queue Start
   /
  /
1       Mehmet Emre Ayar
  0
 /|\
 /.\
2       Gulsum Ikbal Avsar
  0
 /|\
 /.\
3       Kemalettin Kocyigit
  0
 /|\
 /.\
4       Rene Magritte
  0
 /|\
 /.\
```

```
Time ===>Morning!!!
Customer Name : Mehmet Emre Ayar
Mehmet Emre Ayar wants to buy '10 Suc ve Ceza'
Suc ve Ceza is available...

                                =
                               ======
                              ============     _____
                             ==================  |   |
                            =======================|   |
                           ========================|   |
                          ============================|   |==
                        ==========================================
                      ================================================
                   ==================  My Book Store  ===================
============================================================
||   Kurk Mantolu Madonna     ||     $19.90     ||   2    ||
||   Son Ada                  ||     $9.90      ||   3    ||
||   Nutuk                    ||     $49.90     ||   20   ||
||   Suc ve Ceza              ||     $9.90      ||   30   ||
||   Gurur ve Onyargi         ||     $29.90     ||   30   ||
||   Ermis                    ||     $9.90      ||   30   ||
============================================================
     /\
     /
    / Queue Start
   /
  /
1       Gulsum Ikbal Avsar
  0
 /|\
 /.\
2       Kemalettin Kocyigit
  0
 /|\
 /.\
3       Rene Magritte
  0
 /|\
 /.\
```

1

```
Customer Name : Gulsum Ikbal Avsar
Gulsum Ikbal Avsar wants to buy '5 Kurk Mantolu Madonna'
Kurk Mantolu Madonna is available...
The book that trying to be bought cannot be supplied. There is not enough book to sale.

                                =
                               ======
                              ============     _____
                             ==================  |   |
                            =======================|   |
                           ========================|   |
                          ============================|   |==
                        ==========================================
                      ================================================
                   ==================  My Book Store  ===================
============================================================
||   Kurk Mantolu Madonna     ||     $19.90     ||   2    ||
||   Son Ada                  ||     $9.90      ||   3    ||
||   Nutuk                    ||     $49.90     ||   20   ||
||   Suc ve Ceza              ||     $9.90      ||   30   ||
||   Gurur ve Onyargi         ||     $29.90     ||   30   ||
||   Ermis                    ||     $9.90      ||   30   ||
============================================================
     /\
     /
    / Queue Start
   /
  /
1       Kemalettin Kocyigit
  0
 /|\
 /.\
2       Rene Magritte
  0
 /|\
 /.\
3       Jason Statham
  0
 /|\
 /.\
```

2

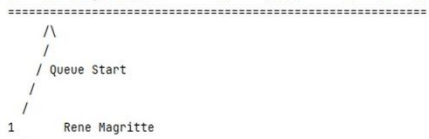
```
Customer Name : Kemalettin Kocyigit
Kemalettin Kocyigit wants to buy '2 Dune'
Dune is not available...The book you were looking for was not found.
                                =
                               ======
                              ============     _____
                             ==================  |   |
                            =======================|   |
                           ========================|   |
                          ============================|   |==
                        ==========================================
                      ================================================
                   ==================  My Book Store  ===================
============================================================
||   Kurk Mantolu Madonna     ||     $19.90     ||   2    ||
||   Son Ada                  ||     $9.90      ||   3    ||
||   Nutuk                    ||     $49.90     ||   20   ||
||   Suc ve Ceza              ||     $9.90      ||   30   ||
||   Gurur ve Onyargi         ||     $29.90     ||   30   ||
||   Ermis                    ||     $9.90      ||   30   ||
============================================================
     /\
     /
    / Queue Start
   /
  /
1       Rene Magritte
  0
 /|\
 /.\
2       Jason Statham
  0
 /|\
 /.\
3       Christopher Nolan
  0
 /|\
 /.\
```

3

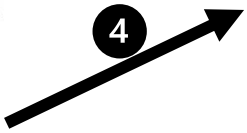
```
Deliver Time!!
Deliver contain:
- Book Name: Dune , Price: $12 , Amount: 15
- Book Name: The Hunger Games , Price: $7.90 , Amount: 5
- Book Name: Ermis , Price: $9.90 , Amount: 7
Dune is not available...The Hunger Games is not available...Ermis is available...
```

```
  My Book Store
```

Kurk Mantolu Madonna	$19.90	2
Son Ada	$9.90	3
Nutuk	$49.90	20
Suc ve Ceza	$9.90	30
Gurur ve Onyargi	$29.90	30
Ermis	$9.90	37
Dune	$12	15
The Hunger Games	$7.90	5



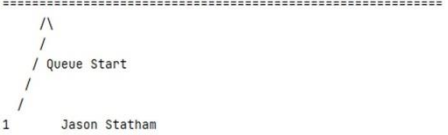
```
Queue Start
1       Rene Magritte
2       Jason Statham
```



```
Time ===>Afternoon!!!
Customer Name : Rene Magritte
Rene Magritte wants to buy '1 Son Ada'
Son Ada is available...
```

```
  My Book Store
```

Kurk Mantolu Madonna	$19.90	2
Son Ada	$9.90	2
Nutuk	$49.90	20
Suc ve Ceza	$9.90	30
Gurur ve Onyargi	$29.90	30
Ermis	$9.90	37
Dune	$12	15
The Hunger Games	$7.90	5



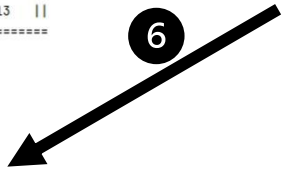
```
Queue Start
1       Jason Statham
2       Christopher Nolan
```

```
Customer Name : Christopher Nolan
Christopher Nolan wants to buy '2 Dune'
Dune is available...
```

```
  My Book Store
```

Kurk Mantolu Madonna	$19.90	2
Son Ada	$9.90	2
Nutuk	$49.90	20
Suc ve Ceza	$9.90	30
Gurur ve Onyargi	$29.90	30
Ermis	$9.90	37
Dune	$12	13

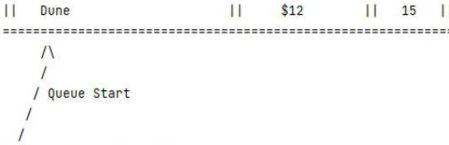
```
Queue Start
1       Stanley Kubrick
2       David Fincher
3       Ted Mosby
```



```
Customer Name : Jason Statham
Jason Statham wants to buy '5 The Hunger Games'
The Hunger Games is available...
The last book has been sold...
```

```
  My Book Store
```

Kurk Mantolu Madonna	$19.90	2
Son Ada	$9.90	2
Nutuk	$49.90	20
Suc ve Ceza	$9.90	30
Gurur ve Onyargi	$29.90	30
Ermis	$9.90	37
Dune	$12	15



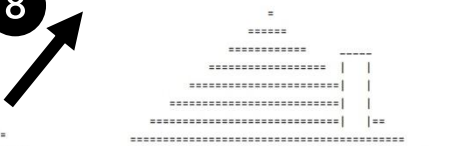
```
Queue Start
1       Christopher Nolan
2       Stanley Kubrick
3       David Fincher
```

```
Time ===>Morning!!!
Customer Name : Stanley Kubrick
Stanley Kubrick wants to buy '10 Suc ve Ceza'
Suc ve Ceza is available...
```

```
  ================== My Book Store ==================
============================================================
||  Kurk Mantolu Madonna    ||    $19.90    ||  2  ||
||  Son Ada                 ||    $9.90     ||  2  ||
||  Nutuk                   ||    $49.90    ||  20 ||
||  Suc ve Ceza             ||    $9.90     ||  20 ||
||  Gurur ve Onyargi        ||    $29.90    ||  30 ||
||  Ermis                   ||    $9.90     ||  37 ||
||  Dune                    ||    $12       ||  13 ||
============================================================
    /\
    /
   / Queue Start
  /
 /
1      David Fincher
 0
/|\
/.\
2      Ted Mosby
 0
/|\
/.\
3      Chandler Bing
 0
/|\
/.\
```

8

```
Customer Name : David Fincher
David Fincher wants to buy '5 Kurk Mantolu Madonna'
Kurk Mantolu Madonna is available...
The book that trying to be bought cannot be supplied. There is not enough book to sale.
```



```
  ================== My Book Store ==================
============================================================
||  Kurk Mantolu Madonna    ||    $19.90    ||  2  ||
||  Son Ada                 ||    $9.90     ||  2  ||
||  Nutuk                   ||    $49.90    ||  20 ||
||  Suc ve Ceza             ||    $9.90     ||  20 ||
||  Gurur ve Onyargi        ||    $29.90    ||  30 ||
||  Ermis                   ||    $9.90     ||  37 ||
||  Dune                    ||    $12       ||  13 ||
============================================================
    /\
    /
   / Queue Start
  /
 /
1      Ted Mosby
 0
/|\
/.\
2      Chandler Bing
 0
/|\
/.\
```

9

```
Customer Name : Ted Mosby
Ted Mosby wants to buy '2 Dune'
Dune is available...
```

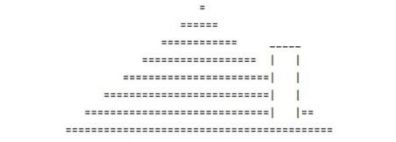
```
  ================== My Book Store ==================
============================================================
||  Kurk Mantolu Madonna    ||    $19.90    ||  2  ||
||  Son Ada                 ||    $9.90     ||  2  ||
||  Nutuk                   ||    $49.90    ||  20 ||
||  Suc ve Ceza             ||    $9.90     ||  20 ||
||  Gurur ve Onyargi        ||    $29.90    ||  30 ||
||  Ermis                   ||    $9.90     ||  37 ||
||  Dune                    ||    $12       ||  11 ||
============================================================
    /\
    /
   / Queue Start
  /
 /
1      Chandler Bing
 0
/|\
/.\
```

10

```
Deliver Time!!
Deliver contain:
- Book Name: Dune , Price: $12 , Amount: 15
- Book Name: The Hunger Games , Price: $7.90 , Amount: 5
- Book Name: Ermis , Price: $9.90 , Amount: 7
Dune is available...
The Hunger Games is not available...Ermis is available...
```

```
  ================== My Book Store ==================
============================================================
||  Kurk Mantolu Madonna    ||    $19.90    ||  2  ||
||  Son Ada                 ||    $9.90     ||  2  ||
||  Nutuk                   ||    $49.90    ||  20 ||
||  Suc ve Ceza             ||    $9.90     ||  20 ||
||  Gurur ve Onyargi        ||    $29.90    ||  30 ||
||  Ermis                   ||    $9.90     ||  44 ||
||  Dune                    ||    $12       ||  26 ||
||  The Hunger Games        ||    $7.90     ||  5  ||
============================================================
    /\
    /
   / Queue Start
  /
 /
1      Chandler Bing
 0
/|\
/.\
```

```
Time ===>Afternoon!!!
Customer Name : Chandler Bing
Chandler Bing wants to buy '1 Son Ada'
Son Ada is available...
```



```
  ================== My Book Store ==================
============================================================
||  Kurk Mantolu Madonna    ||    $19.90    ||  2  ||
||  Son Ada                 ||    $9.90     ||  1  ||
||  Nutuk                   ||    $49.90    ||  20 ||
||  Suc ve Ceza             ||    $9.90     ||  20 ||
||  Gurur ve Onyargi        ||    $29.90    ||  30 ||
||  Ermis                   ||    $9.90     ||  44 ||
||  Dune                    ||    $12       ||  26 ||
||  The Hunger Games        ||    $7.90     ||  5  ||
============================================================
    /\
    /
   / Queue Start
  /
 /

Customer Name :
Process finished with exit code -1073741819 (0xC0000005)
```