# HACETTEPE UNIVERSITY

# DEPARTMENT OF COMPUTER ENGINEERING

## BBM103 ASSIGMENT 4 REPORT

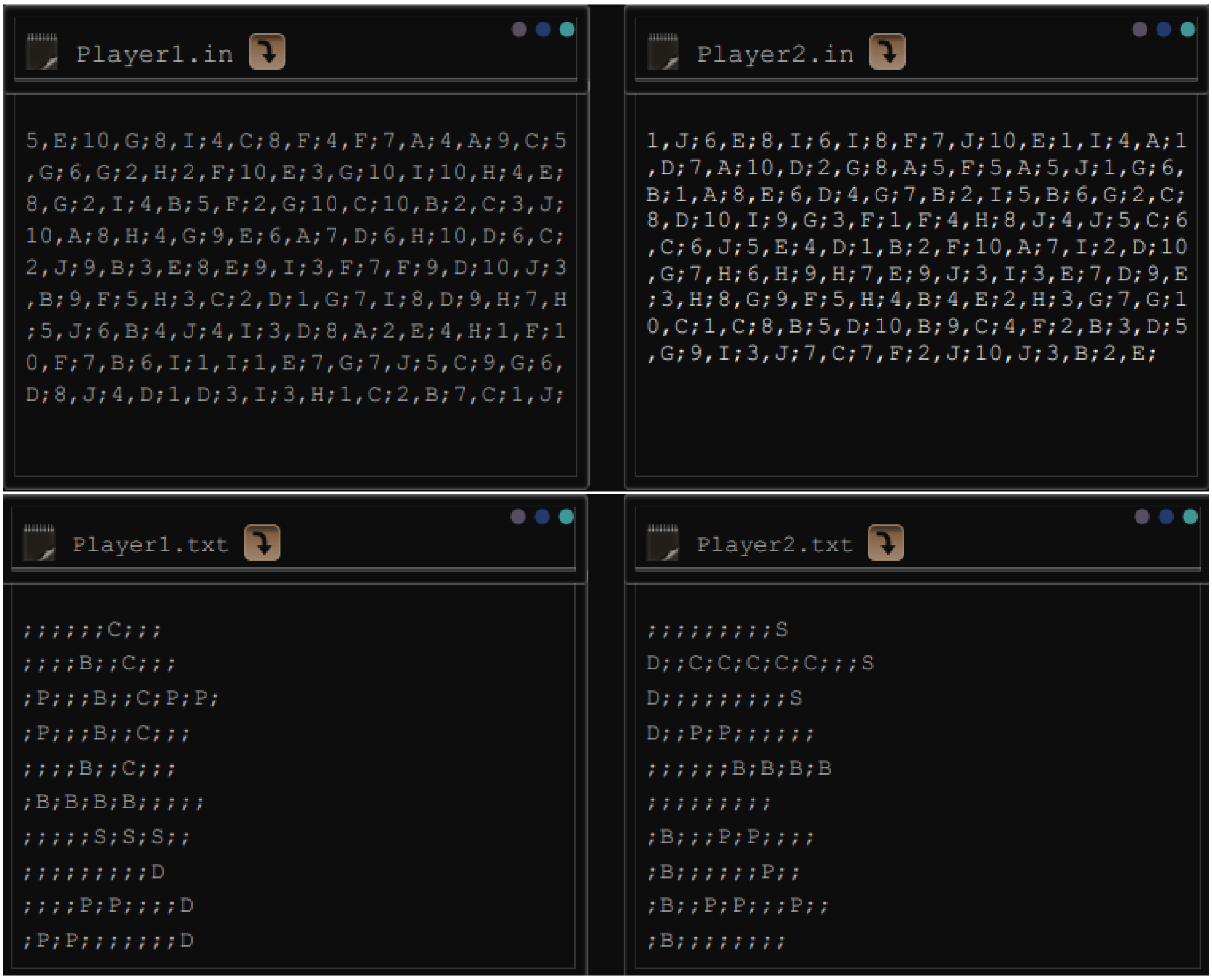
İLHAN ARAS- 2210356023

03.01.2023

# Analysis

The main purpose is making battleship game in this assignment with using file IO.

Given input files Player1.txt Player2.txt Player2.in Player1.in.



```
5,E;10,G;8,I;4,C;8,F;4,F;7,A;4,A;9,C;5
,G;6,G;2,H;2,F;10,E;3,G;10,I;10,H;4,E;
8,G;2,I;4,B;5,F;2,G;10,C;10,B;2,C;3,J;
10,A;8,H;4,G;9,E;6,A;7,D;6,H;10,D;6,C;
2,J;9,B;3,E;8,E;9,I;3,F;7,F;9,D;10,J;3
,B;9,F;5,H;3,C;2,D;1,G;7,I;8,D;9,H;7,H
;5,J;6,B;4,J;4,I;3,D;8,A;2,E;4,H;1,F;1
0,F;7,B;6,I;1,I;1,E;7,G;7,J;5,C;9,G;6,
D;8,J;4,D;1,D;3,I;3,H;1,C;2,B;7,C;1,J;
```

```
1,J;6,E;8,I;6,I;8,F;7,J;10,E;1,I;4,A;1
,D;7,A;10,D;2,G;8,A;5,F;5,A;5,J;1,G;6,
B;1,A;8,E;6,D;4,G;7,B;2,I;5,B;6,G;2,C;
8,D;10,I;9,G;3,F;1,F;4,H;8,J;4,J;5,C;6
,C;6,J;5,E;4,D;1,B;2,F;10,A;7,I;2,D;10
,G;7,H;6,H;9,H;7,E;9,J;3,I;3,E;7,D;9,E
;3,H;8,G;9,F;5,H;4,B;4,E;2,H;3,G;7,G;1
0,C;1,C;8,B;5,D;10,B;9,C;4,F;2,B;3,D;5
,G;9,I;3,J;7,C;7,F;2,J;10,J;3,B;2,E;
```

```
;;;;;;C;;;
;;;;B;;C;;;
;P;;;B;;C;P;P;
;P;;;B;;C;;;
;;;;B;;C;;;
;B;B;B;B;;;;;
;;;;;S;S;S;;
;;;;;;;;;D
;;;;P;P;;;;D
;P;P;;;;;;;D
```

```
;;;;;;;;;S
D;;C;C;C;C;C;;;S
D;;;;;;;;;S
D;;P;P;;;;;;
;;;;;;B;B;B;B
;;;;;;;;;
;B;;;P;P;;;;
;B;;;;;;P;;
;B;;P;P;;;P;;
;B;;;;;;;;
```

First we read the txt files and create the board of the game, then we read the .in files respectively and make the shots.

# Programmer's Catalogue

## 1)Saving function

```
def saving_func(inpt):#save messages into text file
   with open('Battleship.out.txt', "a",encoding="utf-8")as f:
        f.write(inpt)
        f.close()
```

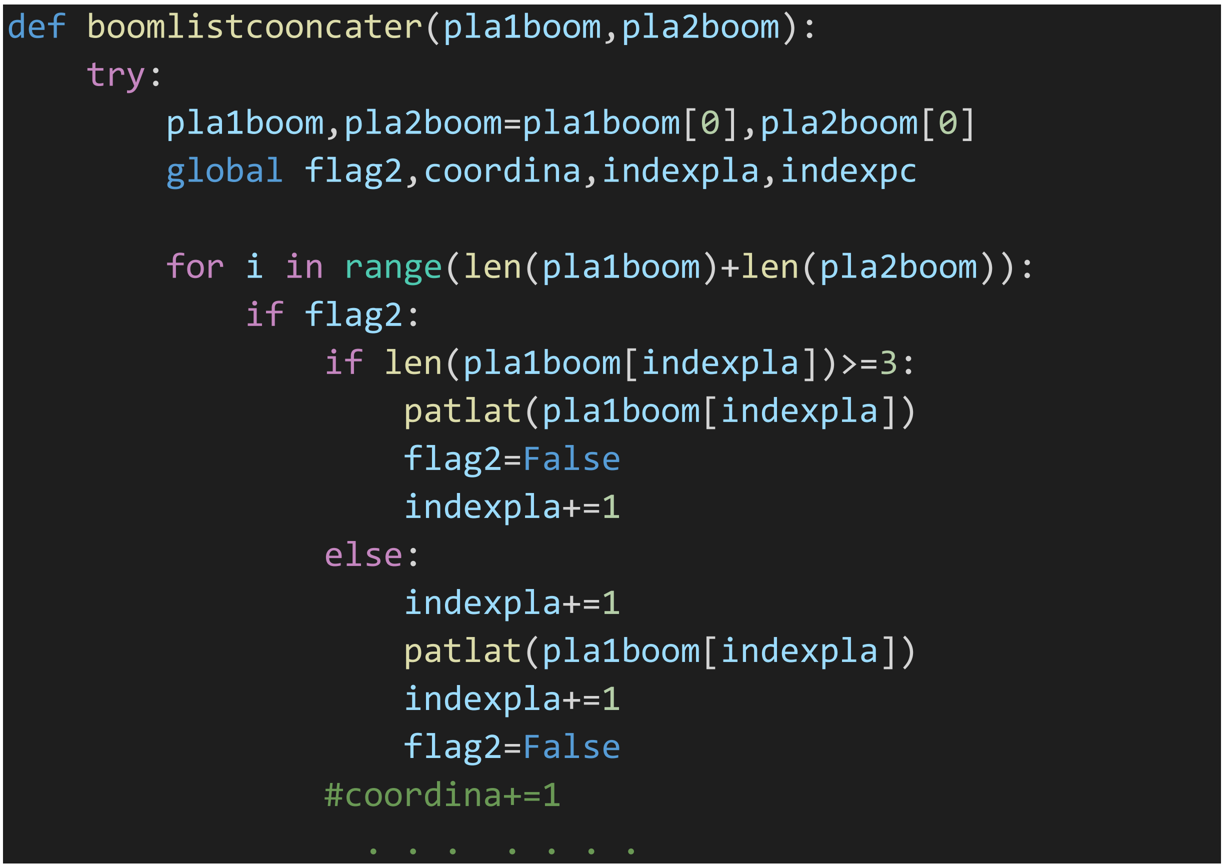
this func saves and writes output in the Battleshi.out.txt file.

## 2)Hitting function

```
def patlat(coordinate):
    global
flag2,listforpc,listforpla,playercizgilist,pccizgilist,indexp
la,indexpc
    global
player1gameorder,player2gameorder,roundcount,hittedplayer1
    rowcoor=int(coordinate.split(",")[0])-1
    colcoor=int(ord(coordinate.split(",")[1])-65)

    try:
        if flag2:
            if player2gameorder[rowcoor][colcoor]=="-":
                pccizgilist[rowcoor][colcoor]="O"
            elif player2gameorder[rowcoor][colcoor]!="-":
                pccizgilist[rowcoor][colcoor]="X"
                colcoor=colcoor+1
                hittedpc.append(colcoor+rowcoor*10)
        else:
            if player1gameorder[rowcoor][colcoor]=="-":
                playercizgilist[rowcoor][colcoor]="O"

            elif player1gameorder[rowcoor][colcoor]!="-":
                playercizgilist[rowcoor][colcoor]="X"


                 . . . . .
```

This function takes the hitting coordinate and replace from "-" to "O" or "X"

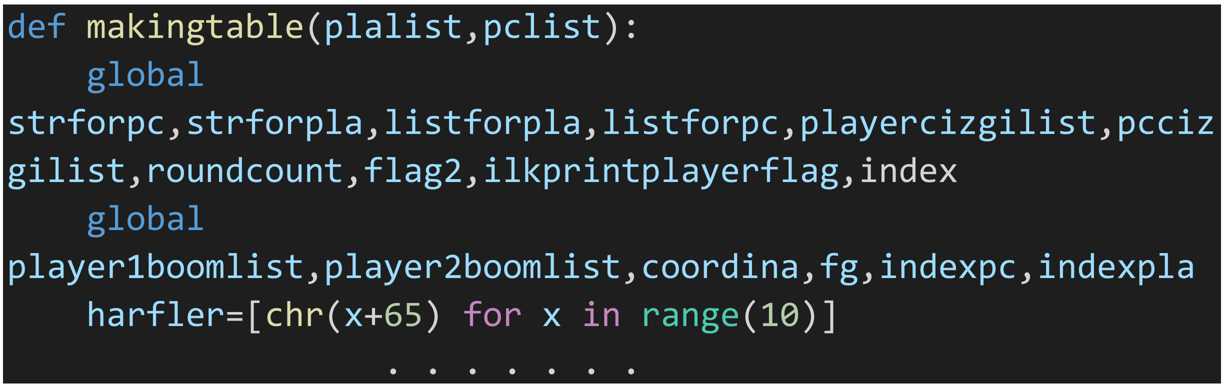
### 3)Boomlistconcater function



```
def boomlistcooncater(pla1boom,pla2boom):
    try:
        pla1boom,pla2boom=pla1boom[0],pla2boom[0]
        global flag2,coordina,indexpla,indexpc

        for i in range(len(pla1boom)+len(pla2boom)):
            if flag2:
                if len(pla1boom[indexpla])>=3:
                    patlat(pla1boom[indexpla])
                    flag2=False
                    indexpla+=1
                else:
                    indexpla+=1
                    patlat(pla1boom[indexpla])
                    indexpla+=1
                    flag2=False
                #coordina+=1
                  . . .  . . . .
```

This function sets the player turns and makes the patlat function work.

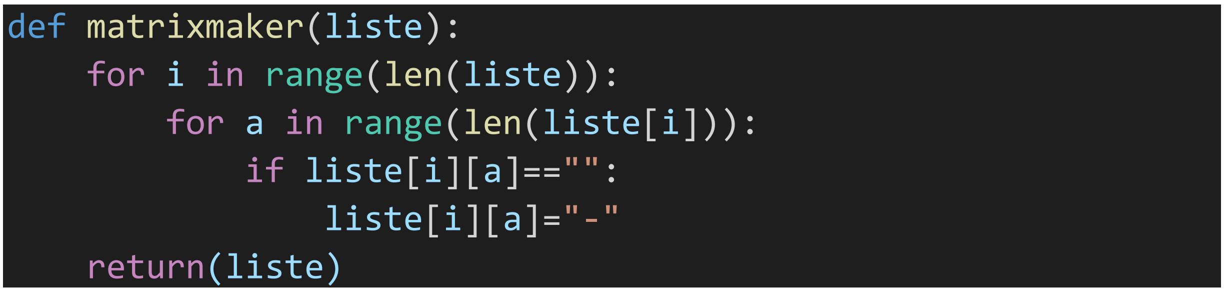
## 4)Makingtable function



```
def makingtable(plalist,pclist):
    global
strforpc,strforpla,listforpla,listforpc,playercizgilist,pcciz
gilist,roundcount,flag2,ilkprintplayerflag,index
    global
player1boomlist,player2boomlist,coordina,fg,indexpc,indexpla
    harfler=[chr(x+65) for x in range(10)]
                   . . . . . . .
```

This func making table and some output strings.

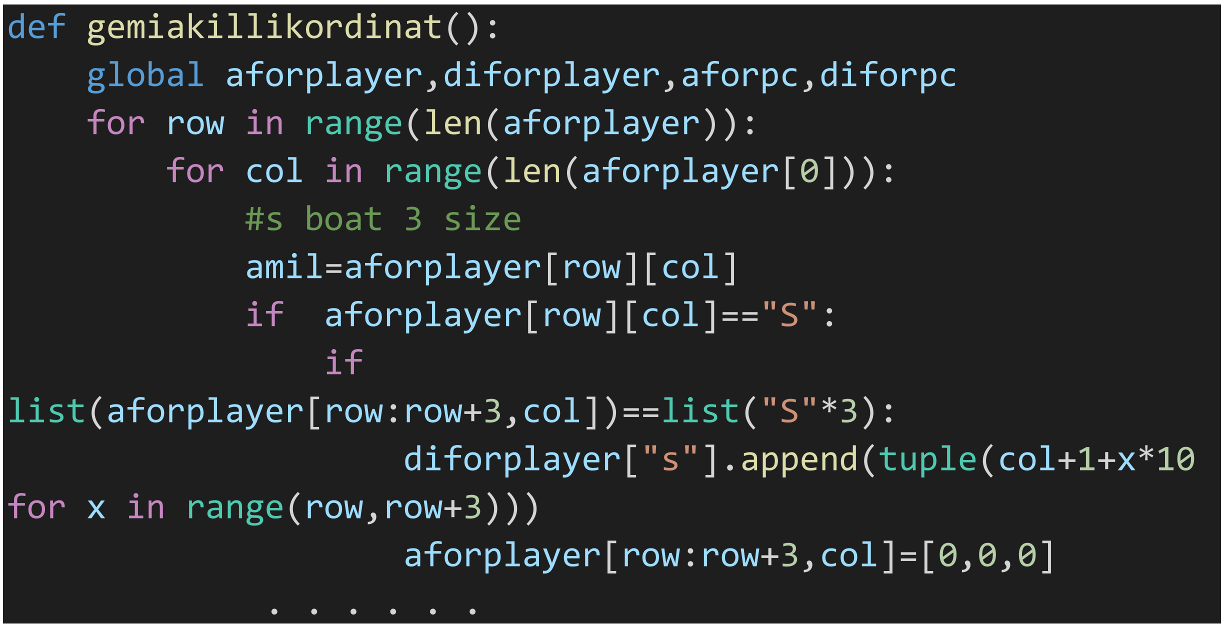
## 5)Matrixmaker function



```
def matrixmaker(liste):
    for i in range(len(liste)):
        for a in range(len(liste[i])):
            if liste[i][a]=="":
                liste[i][a]="-"
    return(liste)
```

this func makes matrix table with “-“

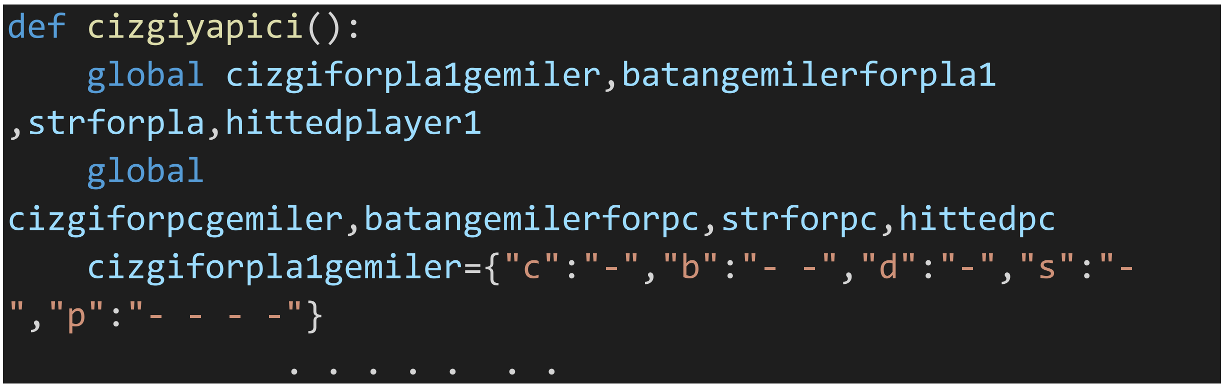
## 6)Gemiakillikordinat func



```
def gemiakillikordinat():
    global aforplayer,diforplayer,aforpc,diforpc
    for row in range(len(aforplayer)):
        for col in range(len(aforplayer[0])):
            #s boat 3 size
            amil=aforplayer[row][col]
            if  aforplayer[row][col]=="S":
                if
list(aforplayer[row:row+3,col])==list("S"*3):
                    diforplayer["s"].append(tuple(col+1+x*10
for x in range(row,row+3)))
                    aforplayer[row:row+3,col]=[0,0,0]
             . . . . . .
```

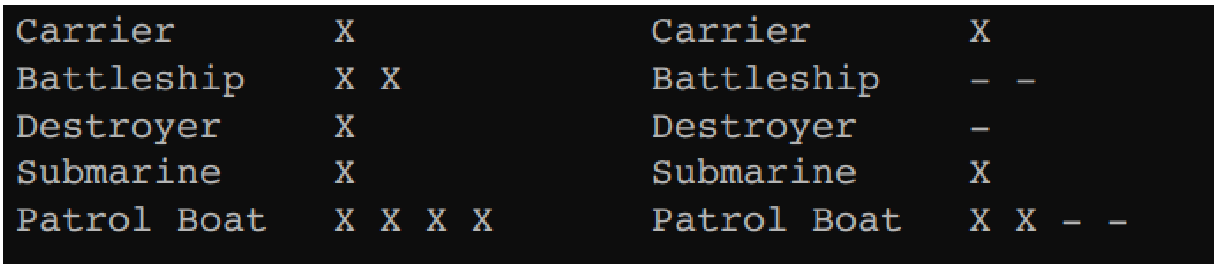
This func find all ships coordinate.

### 7)Cizgiyapici func



```
def cizgiyapici():
    global cizgiforpla1gemiler,batangemilerforpla1
,strforpla,hittedplayer1
    global
cizgiforpcgemiler,batangemilerforpc,strforpc,hittedpc
    cizgiforpla1gemiler={"c":"-","b":"- -","d":"-","s":"-
","p":"- - - -"}
              . . . . .  . .
```

this func puts “-“ in the dict for that



```
Carrier         X                   Carrier         X
Battleship      X X                 Battleship      - -
Destroyer       X                   Destroyer       -
Submarine       X                   Submarine       X
Patrol Boat     X X X X             Patrol Boat     X X - -
```

## Design

Firstly, program reads .txt and .in file. Program puts values from input files to lists.

Matrixmaker func splits “;” after some initiliaze values gemiakillikordinat func finds all coordinates of ships. Makingtable func makes 1. Table and than boomlistcooncater func works and than patlat func works. Patlat func makes “-“ to “X” or “O” than makingtable func works writes some text in output file after that cizgiyapici func works to

create dict which includes some curutial informations and appending some strings into strforpla and strforpc. After that proccess program write table and move information strings.

# User Catalogue

The user will enter the locations of ships using “;” in .txt files.

The user will enter hit coordinates using with ";" and "," in .in files

for example 5,E;10,G;8,I;4,C;8,F;4,F;7,A;4,A;9,C; ...

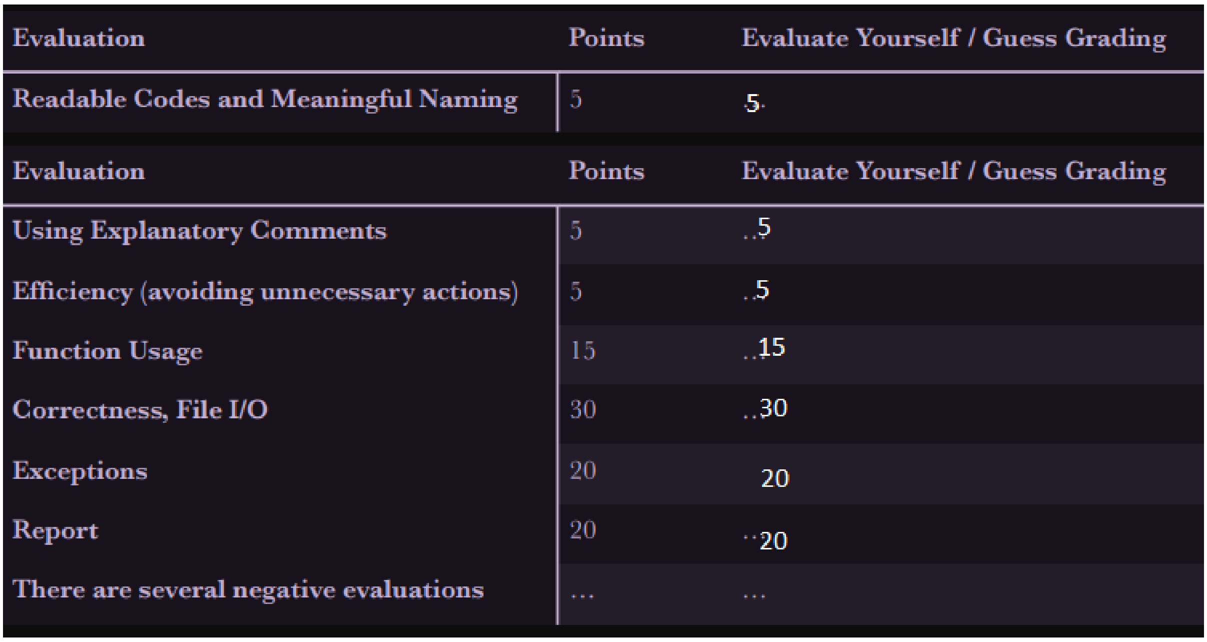
Restrictions: if values are missed, program return some error messages

IOError: A problem can occur if the input files are not reachable at the specified file path (if files do not exist, or if their names are misspelled, etc.).

IndexError: For missing arguments, your program should throw an index error and handle it by giving an explanatory prompt to the user. For example, “,A;” “A;” “,;” “;” “1,;” “1;”

ValueError: If any of the first operands are given as non-numeric values, trying to convert them to integers will raise a ValueError exception. Also, if operands are given but you can not interpret them, you will raise the ValueError exception. For example, “A,1;” “1,1;” “A,A;” “5,E10,G;”

AssertionError:for example “11,A;” “5,K;” causes that error.



| Evaluation | Points | Evaluate Yourself / Guess Grading |
|---|---|---|
| Readable Codes and Meaningful Naming | 5 | 5 |

| Evaluation | Points | Evaluate Yourself / Guess Grading |
|---|---|---|
| Using Explanatory Comments | 5 | ..5 |
| Efficiency (avoiding unnecessary actions) | 5 | ..5 |
| Function Usage | 15 | ..15 |
| Correctness, File I/O | 30 | ..30 |
| Exceptions | 20 | 20 |
| Report | 20 | ..20 |
| There are several negative evaluations | ... | ... |