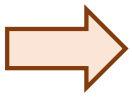
ASSIGNMENT #3

SUBJECT & BASIC INFORMATION

- Writing down a C++ automation program that performs grocery operations according to the criteria stated below.
  - The program has two main functionality: assigning new products to the list (max size is 10) and printing a the product list on the screen
  - There are 4 different product type in the supermarket as **"1-FOOD"**, **"2-DRINK"**, **"3-JUNK"** and **"4-STATIONARY"**.
  - Taxes depending on the product types are 3%, 5%,18% and 8% respectivly.

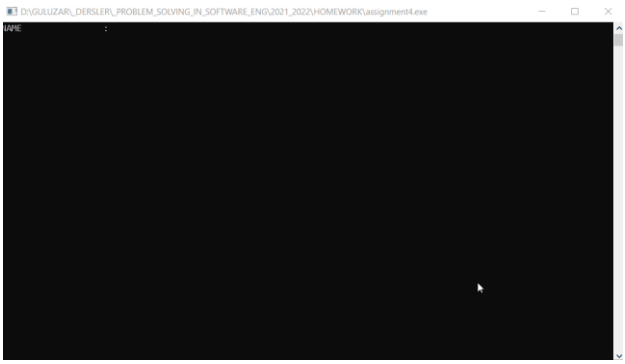
SAMPLE SCREEN OUTPUT

```
NAME                : AZRA
SURNAME             : CIT
ID                  : 2015
DEPARTMENT          : 2
TOTAL WORKING HOURS : 50
HOURLY WAGE         : 40
--------------------------------------
NAME                : RANA
SURNAME             : CIT
ID                  : 2008
DEPARTMENT          : 3
NUMBER OF PIECES    : 100
PIECE PRICE         : 10
PREMIUM (%)         : 8
```



```
----------------------------------------
FULL NAME           : AYSE METIN
ID                  : 1001
DEPARTMENT          : MANAGEMENT
WEEKLY SALARY       : 1000 TL
----------------------------------------
FULL NAME           : AZRA CIT
ID                  : 2015
DEPARTMENT          : LOGISTICS
WEEKLY SALARY       : 1200 TL
----------------------------------------
FULL NAME           : RANA CIT
ID                  : 2008
DEPARTMENT          : MANUFACTURING
WEEKLY SALARY       : 580 TL
Press any key to continue . . .
```

SAMPLE PROGRAM VIDEO



REQUIREMENTS

- Define an enumeration type named **CATEGORY** that keeps the type of the product
- Define a struct named **PRODUCT** that keeps the information of the products (**name, id, type** and **price**)
- Define the type of the **PRODUCT** struct as the **CATEGORY** enumeration type.
- Define an array of 10-elements named **products** in **PRODUCT** struct to hold the product information
- The value of the first four element of the **products** array is initially assigned as follows:

NAME	ID	CATEGORY	PRICE
BREAD	1	FOOD	4
AYRAN	2	DRINK	4
CHIPS	3	JUNK	9
PENCIL	5	STATIONARY	20

- Other elements of the **employee** array are requested from the user.
  - Write a function named **addProduct()**
  - The **addProduct()** function will ask the user for the product's name, id, category and price and assigns these values to an **PRODUCT** struct variable.
  - The **addProduct()** function will return the relevant product information which are assigned to an **PRODUCT** struct variable
  - Write a function named **print()**
  - The **printProducts()** method writes all the information of the **products** array on the screen a
  - The **printProducts()** method will get the **products** array as input parameter.
  - The **printProducts()** method will calculate tax information of each product of the array

RULES & EVALUATION

- Using a **goto** statement is strictly prohibited.
- Each C++ file should include this comment lines below at the beginning of the C++ file

```
//**************************************************************************
//**                                                                      **
//**         STUDENT NAME.............:                                   **
//**         STUDENT NUMBER...........:                                   **
//**************************************************************************
```

- You should compile your codes with MingGW or GCC. (NOTE: If you use another compiler, please test your codes with these compilers before uploading your homework on system)
- **Deadline:** Control SABIS system
- A report should be prepared for each assignment
  - First page of the report should be a cover page including student information (name, surname, number, lecturer, course name, ...)
  - The content of the assignment (a brief explanation of your program) should be included after the cover page
  - At the end of the report, there should be an **'honor code'** signed by yourself.
- You should upload only your C++ file (.cpp file) and your report (in pdf format) together before deadline.
- Evaluation Criteria
  - Comment lines (student information, explaining operations like variable names, if statements, loops, etc.)
  - Obeying the variable declaration rules
  - Being readable (intendation, comments, etc.)
  - Correct compilation of the code
  - Reporting (cover page, content, honor code, etc.)
  - …