

Payroll System

First we would like to thank ...

- Dr. Mohamed Sameh
- Eng. Heba Elgabri
- Eng. Salma Elgabri

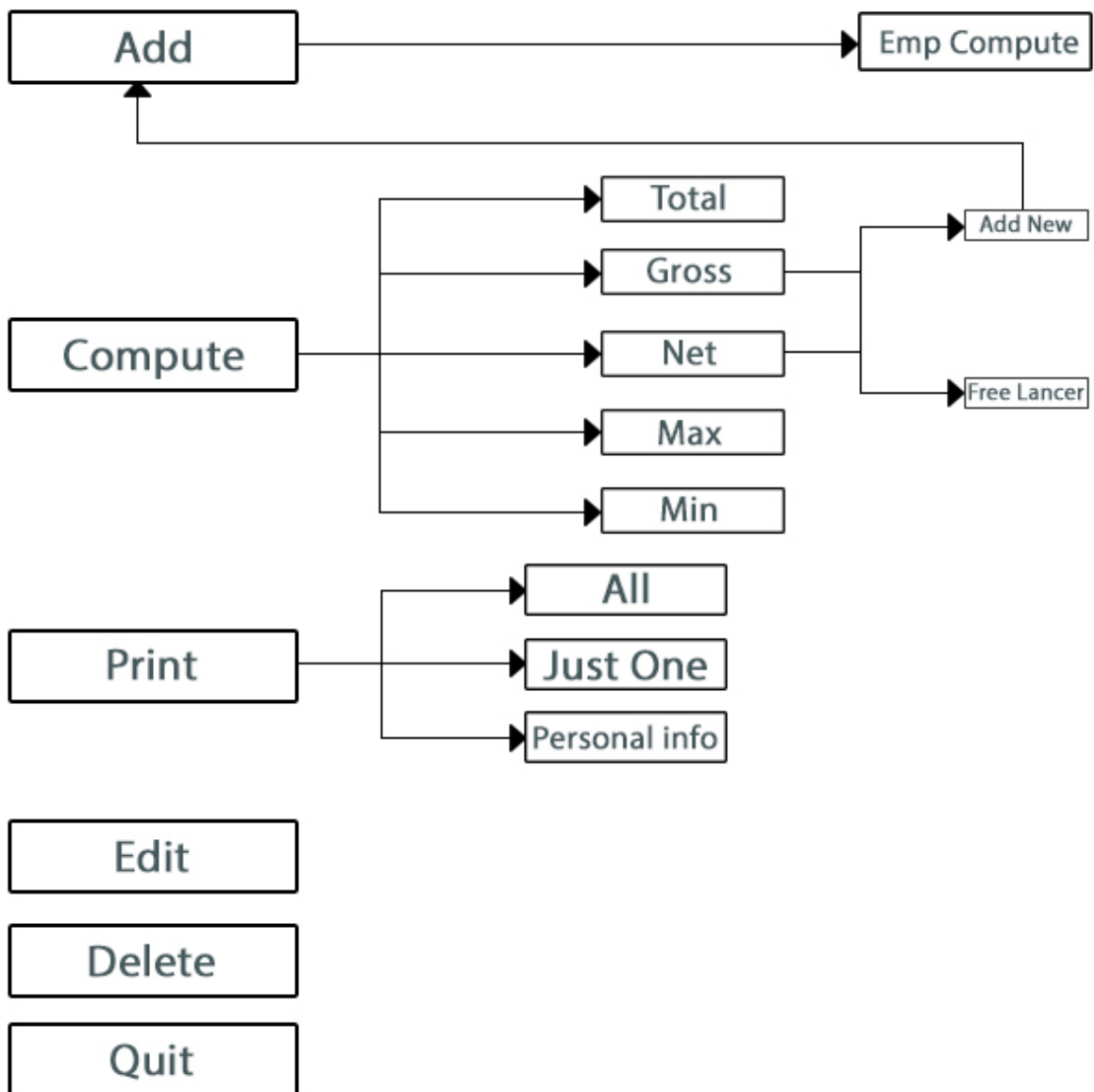
For their efforts and much appreciated help.

Introduction

This program purpose is to calculate the total salary, taxes, and the net salary of employees based on their working hours and pay rate.

It also calculate the total company payroll and the maximum paid and the minimum paid.

Let's start with a simple chart ...



Overview over functions and global variables used!

```
#include ... //setw
using namespace std;
struct employee { ... };
employee emp[10]; //array of struct holding 10 employees data
int emp_count = 0; //counter of employees and used sometimes as index for the array
int emp_no_after_sorting; //holding the index of the last added employee after sorting

void welcome(); //welcome message at the beginning
void verification(); //not anyone can access employees and see their data
void menu(); //main menu of the program and other functions are called within it
void add(); //add another emp function
void add_another_emp(); //asks for adding another emp!
void view_all(); //print id and name of all employees
void print(); // print function with many options
void edit(); //edit a current emp
void erase(); //delete emp
void compute(); //contains secondary menu with many compute options
void receiveF(); //recover data from files
void sendF(); //send data to files
void personal_info(int); //option for adding some personal info
void freelancer_gross(double &F_gross); //calculate gross salary for a free lancer , wont be saved in the array (call by refrence)
void emp_compute(int l); //it calculate the employee gross salary-tax-pay rate-worked hours-net salary ....etc
void sort(int y); //sort employees automatically after adding or editing in the array by descending order
double calc_tax(double); // calculate taxes based on the gross salary
double calc_workedhours(double); //ceil or floor for the worked hours

int main()
{
    menu();

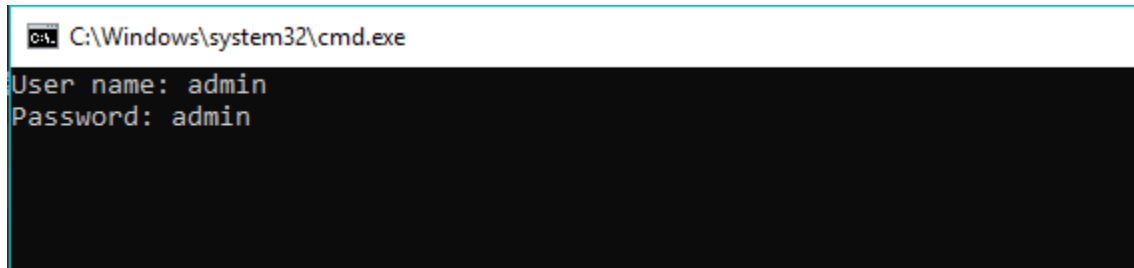
    return 0;
}

void verification() { ... }
void welcome() { ... }
void menu() { ... }
void add() { ... }
void add_another_emp() { ... }
void view_all() { ... }
void print() { ... }
void edit() { ... }
void erase() { ... }
void compute() { ... }
void receiveF() { ... }
void sendF() { ... }
void personal_info(int u) { ... }
void freelancer_gross(double &F_gross) { ... }
void emp_compute(int l) { ... }
void sort(int y) { ... }
double calc_tax(double k) { ... }
double calc_workedhours(double y) { ... }
```

This is how we handled some errors entering character into integer variable or otherwise!

```
while (cin.get() != '\n')
{
    cout << "Error! try again" << endl;
    cin.clear();
    cin.ignore(100, '\n');
    cin >> choice;
}
```

Let's get into the program and try it ...




```
C:\Windows\system32\cmd.exe
User name: admin
Password: admin
```

We added an optional function so not anyone can access employee's database!

If the username or the password entered 3 times wrong the program will terminate.

Welcome!

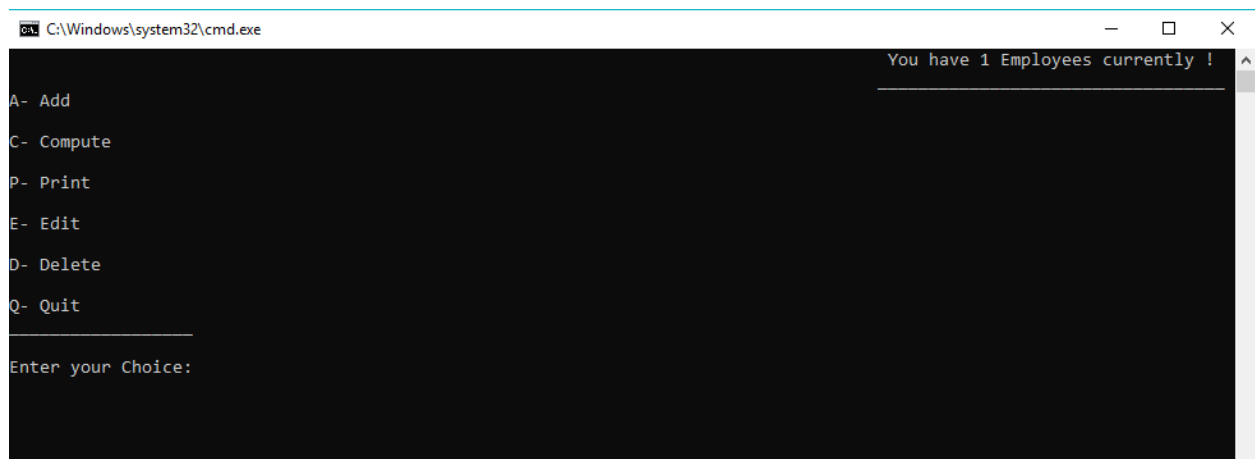


```
C:\Windows\system32\cmd.exe

Welcome to thebes payroll

Press any key to continue . . .
```

Main menu:



```
C:\Windows\system32\cmd.exe

You have 1 Employees currently !

A- Add
C- Compute
P- Print
E- Edit
D- Delete
Q- Quit
-----
Enter your Choice:
```

This is part of the menu code.

```
void menu()
{
    verification();
    welcome();
    receiveF();
    char choice;
    do
    {
        system("cls");
        cout << setw(95) << "You have " << emp_count << " Employees currently !" << endl;
        cout << setw(119) << " _____ " << endl;
        cout << "A- Add \n\n"
             << "C- Compute\n\n"
             << "P- Print \n\n"
             << "E- Edit \n\n"
             << "D- Delete \n\n"
             << "Q- Quit\n"
             << " _____ \n\n"
             << "Enter your Choice: ";

        cin >> choice;
        while (cin.get() != '\n') { ... }

        switch (choice)
        {
            case 'A':
            case 'a':
            {
                add();
                add_another_emp();
                break;
            }
            case 'C':
            case 'c':
            {
                compute();
                break;
            }
            case 'E':
            case 'e':
            {
                edit();
                break;
            }
            case 'P':
            case 'p':
            {
                print();
                break;
            }
        }
    }
}
```

Let's add an employee and compute his salary.

cmd C:\Windows\system32\cmd.exe

First name: mohamed
Last name: ahmed

```
C:\Windows\system32\cmd.exe
Do you want to compute his salary ?
1-Yes
2-No
1
Enter employee pay rate: 150
1- Enter hours worked per month
2- Enter daily hours worked
1
Hours worked this month: 176
```

```
C:\Windows\system32\cmd.exe
Do you want to add personal info ?
1- Yes
2- No
1
Age: 35
Job Description: student
Phone: asd
Error enter numbers only !
Phone: 01235686
Address: 11 el asjdh
E-mail: aksgjhd@gmail.com
```

```
C:\Windows\system32\cmd.exe
Do you want to add another employee ?
1-Yes
2-No
2
```

After adding any employee it asks if u want to compute his salary, it accept entering hours day by day by entering how many days worked or total hours worked this month.

Then asks if u want to add his personal info finally it

Asks for adding another employee. And there's a part of the code.


```

void add()
{
    system("cls");
    if (emp_count > 9)
    {
        cout << "Exceeded employees number delete any before you can add again !" << endl;
        system("pause");
    }
    else
    {
        int choice_1;           //choosing to compute salary or not!
again10:
        cout << "First name: ";           //getting employee first and last name
        getline(cin, emp[emp_count].Fname);
        for (int i = 0; i < emp[emp_count].Fname.length(); i++) //checking if there's a space in the name
        {
            if (emp[emp_count].Fname[i] == ' ')
            {
                cout << "Error dont enter a space!" << endl;
                goto again10;
            }
        }
again11:
        cout << "Last name: ";
        getline(cin, emp[emp_count].Lname);
        for (int i = 0; i < emp[emp_count].Lname.length(); i++) //checking if there's a space in the name
        {
            if (emp[emp_count].Lname[i] == ' ')
            {
                cout << "Error dont enter a space!" << endl;
                goto again11;
            }
        }
again1:
        //if didnt choose 1 or 2 will come here
        system("cls");
        cout << "Do you want to compute his salary ?\n"
              "1-Yes\n"
              "2-No\n";
        cin >> choice_1;
        while (cin.get() != '\n') // handle entering anything except int
        {
            cout << "Error try again ! " << endl;
            cin.clear();
            cin.ignore(100, '\n');
            cout << "Do you want to compute his salary ?\n"
                  "1-Yes\n"
                  "2-No\n";
            cin >> choice_1;
        }
        if (choice_1 == 1) // computing employee salary
        {
            emp_compute(emp_count);
            emp[emp_count].ID = 5001 + emp_count;
            system("cls");
            cout << "Employee Added Successfully" << endl;
            system("pause");
        }
    }
}

```

It automatically sort employees after computing or editing according to their salary from highest to lowest

```
void sort(int y)
{
    for (int i = 0; i < y; i++)
    {
        if (emp[y].net > emp[i].net)
        {
            employee temp = emp[y];
            for (int j = y; j > i; j--)
            {
                emp[j] = emp[j - 1];
            }
            emp[i] = temp;

            emp_no_after_sorting = i;
            break;
        }
        else
            emp_no_after_sorting = emp_count;
    }
    for (int i = emp_count - 1; i > y; i--)
    {
        if (emp[y].net < emp[i].net)
        {
            employee temp = emp[y];
            for (int j = y; j < i; j++)
            {
                emp[j] = emp[j + 1];
            }
            emp[i] = temp;

            emp_no_after_sorting = i;
            break;
        }
        else
            emp_no_after_sorting = emp_count;
    }
}
```

Let's check the print ...

It has 4 options printing all employees or just one or printing an employee personal information and a back to main menu option.

```
C:\Windows\system32\cmd.exe
1- Print all employees
2- print specific employee
3- Print an employee personal data
4- Main menu
```

- Option 1:

```
C:\Windows\system32\cmd.exe
#      ID      Name      Pay rate      Worked hours      Tax      Gross      Net
1      5003     mohamed ahmed     150           176           6473.04     40128     33655
2      5003     ahmed rizk        150           100           2804.88     18468     15663.1
3      5002     ahmed gamal        0             0             0           0           0
Press any key to continue . . .
```

- Option 2:

```
C:\Windows\system32\cmd.exe
#      ID      Name
1      5003     mohamed ahmed
2      5003     ahmed rizk
3      5002     ahmed gamal
~~~~~
Select employee u want to print:
```

By choosing an emp it prints it as following

```
C:\Windows\system32\cmd.exe
ID      Name      Pay rate      Worked hours      Tax      Gross      Net
5003     ahmed rizk        150           100           2804.88     18468     15663.1
Press any key to continue . . .
```

- Option 3:

C:\Windows\system32\cmd.exe

```
Name: mohamed ahmed
Age: 35
Job Describtion: student
Phone: 01235686
Address: 11 el asjdh
E-mail: aksgjhd@gmail.com
Press any key to continue . . .
```

This happens if u choose an emp without any info

C:\Windows\system32\cmd.exe

```
No personal info to view !
Press any key to continue . . .
```

```
system("cls");
int choice;
cout << "1- Print all employees\n"
      "2- print specific employee\n"
      "3- Print an employee personal data\n"
      "4- Main menu\n";
cin >> choice;
while (cin.get() != '\n') { ... }
if (choice == 1)
{
    system("cls");
    cout << "#\tID\tName\t\t\tPay rate\tWorked hours\tTax\tGross\tNet\n" << endl;
    for (int i = 0; i < emp_count; i++)
    {
        for (int j = i; j <= i; j++)
        {
            if (emp[j].Fname.length() + emp[j].Lname.length() > 14)
            {
                cout << i + 1 << "\t" << emp[j].ID << "\t" << emp[j].Fname << " " << emp[j].Lname << "\t" << emp[j].payrate << "\t\t"
                    << emp[j].workedhours << "\t\t" << emp[j].tax << "\t\t" << emp[j].gross << "\t\t" << emp[j].net << endl;
            }
            else if (emp[j].Fname.length() + emp[j].Lname.length() < 7)
            {
                cout << i + 1 << "\t" << emp[j].ID << "\t" << emp[j].Fname << " " << emp[j].Lname << "\t\t" << emp[j].payrate << "\t\t"
                    << emp[j].workedhours << "\t\t" << emp[j].tax << "\t\t" << emp[j].gross << "\t\t" << emp[j].net << endl;
            }
            else
            {
                cout << i + 1 << "\t" << emp[j].ID << "\t" << emp[j].Fname << " " << emp[j].Lname << "\t\t" << emp[j].payrate << "\t\t"
                    << emp[j].workedhours << "\t\t" << emp[j].tax << "\t\t" << emp[j].gross << "\t\t" << emp[j].net << endl;
            }
        }
    }
    cout << endl;
}
system("pause");
system("cls");
}
```

Let's go to the compute menu!

```
C:\Windows\system32\cmd.exe
T- Total      G- Gross      N- Net      X- Max      I- Min      M- Main menu
Enter your Choice:
```

Case T:

```
C:\Windows\system32\cmd.exe
Total company payroll = 58596
Press any key to continue . . .
```

Case G:

```
C:\Windows\system32\cmd.exe
1- Add a new employee and calculate its gross salary
2- Calculate gross salary for a free lancer (wont be saved)
```

Let's try the 2nd option since we already used the add function

```
C:\Windows\system32\cmd.exe
Enter employee pay rate: 150
1- Enter hours worked per month
2- Enter daily hours worked
1
Hours worked this month: 100
Gross = 15000
Press any key to continue . . .
```

Same For Net function!

Case X:

```
C:\Windows\system32\cmd.exe
Max Salary= 33655 Payed For: mohamed ahmed
Press any key to continue . . .
```

Case I:

```
C:\Windows\system32\cmd.exe
Min Salary= 0 Payed For: ahmed gamal
Press any key to continue . . .
```

Part of the code:

```

void compute()
{
    system("cls");
    char choice_1;
    int choice_2;
again4:
    cout <<
        "T- Total" << setw(20) <<
        "G- Gross" << setw(20) <<
        "N- Net" << setw(20) <<
        "X- Max" << setw(20) <<
        "I- Min" << setw(20) <<
        "M- Main menu" << setw(20) << endl;
    cout << "-----\n"
        "Enter your Choice: ";
    cin >> choice_1;
    while (cin.get() != '\n') { ... }
    switch (choice_1)
    {
    case 'T':
    case 't':
    {
        system("cls");
        double total = 0;
        for (int i = 0; i < emp_count; i++)
        {
            total += (emp[i].net + emp[i].tax);
        }
        cout << "Total company payroll = " << total << endl;
        system("pause");
        system("cls");
        break;
    }
    case 'G':
    case 'g':
    {
again5:
        system("cls");
        cout << "1- Add a new employee and calculate its gross salary\n"
            "2- Calculate gross salary for a free lancer (wont be saved)" << endl;
        cin >> choice_2;
        while (cin.get() != '\n') { ... }
        if (choice_2 == 1) { ... }
        else if (choice_2 == 2) { ... }
        else { ... }
        break;
    }
    }
}

```

Edit!

```
C:\Windows\system32\cmd.exe
#      ID      Name
1      5003     mohamed ahmed
2      5003     ahmed rizk
3      5002     ahmed gamal
~ ~ ~ ~ ~
Select employee u want to edit:
```

By selecting any of them...

```
C:\Windows\system32\cmd.exe
First Name: asdas
Last Name: iauhd
Enter employee pay rate: 123
1- Enter hours worked per month
2- Enter daily hours worked
1
Hours worked this month: 100
```

```
C:\Windows\system32\cmd.exe
Do you want to add personal info ?
1- Yes
2- No
2
Employee edited successfully
Press any key to continue . . .
```



```

void edit()
{
    if (emp_count == 0)
    {
        system("cls");
        cout << "No employees to Edit " << endl;
        system("pause");
        system("cls");
    }
    else
    {
        try2:
        system("cls");
        view_all();
        int x; // choose emp to edit
        cout << "Select employee u want to edit: ";
        cin >> x;
        while (cin.get() != '\n')
        {
            cout << "Error try again ! " << endl;
            cin.clear();
            cin.ignore(100, '\n');
            cout << "Select employee u want edit: ";
            cin >> x;
        }
        if (x > emp_count || x <= 0)
        {
            cout << "Error! Out of range, try again" << endl;
            system("pause");
            goto try2;
        }
        else
        {
            system("cls");
            cout << "First Name: ";
            getline(cin, emp[x - 1].Fname);
            cout << "Last Name: ";
            getline(cin, emp[x - 1].Lname);
            emp_compute(x - 1);
            personal_info(x - 1);
            cout << "Employee edited successfully" << endl;
            system("pause");
            sort(x - 1);
            sendF();
        }
    }
}

```

Delete.

```
C:\Windows\system32\cmd.exe

#      ID      Name
1      5003     mohamed ahmed
2      5003     asdas iauhd
3      5002     ahmed gamal
~~~~~~
Select employee u want to delete:
```

```
C:\Windows\system32\cmd.exe

Are you sure you want to delete ahmed gamal ?
1- Yes
2- No
1
Employee deleted Successfully
Press any key to continue . . .
```

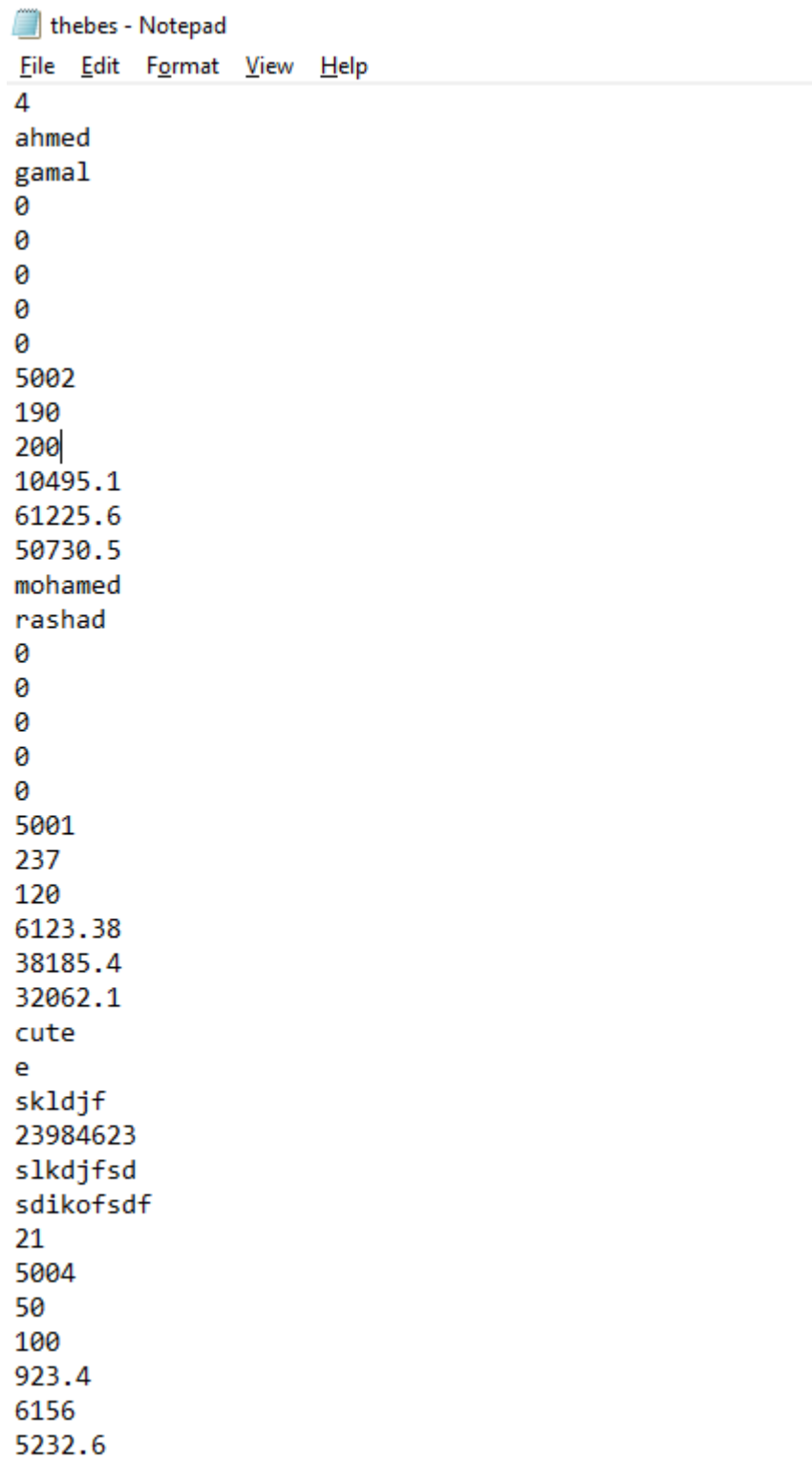
```
for (int i = (x - 1); i < emp_count; i++)
{
    emp[i] = emp[i + 1];
}
cout << "Employee deleted Successfully" << endl;
emp_count--;
system("pause");
system("cls");
```

Quit!

```
C:\Windows\system32\cmd.exe

Thank you for using our system *_*
Press any key to continue . . .
```

Let's take a look on files



thebes - Notepad

File Edit Format View Help

4
ahmed
gamal
0
0
0
0
0
5002
190
200
10495.1
61225.6
50730.5
mohamed
rashad
0
0
0
0
0
5001
237
120
6123.38
38185.4
32062.1
cute
e
skldjf
23984623
slkdjfsd
sdikofsdf
21
5004
50
100
923.4
6156
5232.6

Data is saved to file after adding, editing or deleting!

And is restored when the program runs.

Here is the sending and receiving functions!

```
void receiveF()
{
    ifstream file;
    file.open("thebes.txt");
    file >> emp_count;    //receiving the number of employees first
    for (int i = 0; i < emp_count; i++)    //loop using number of employees to recover all their data
    {
        file >> emp[i].Fname;
        file >> emp[i].Lname;
        file >> emp[i].job;
        file >> emp[i].phone;
        file >> emp[i].address;
        file >> emp[i].mail;
        file >> emp[i].age;
        file >> emp[i].ID;
        file >> emp[i].payrate;
        file >> emp[i].workedhours;
        file >> emp[i].tax;
        file >> emp[i].gross;
        file >> emp[i].net;
    }
    file.close();
}

void sendF()
{
    ofstream file;
    file.open("thebes.txt");
    file << emp_count << endl;    //send number of employees first
    for (int i = 0; i < emp_count; i++)    // sending all employees data to file one by one
    {
        file << emp[i].Fname << endl;
        file << emp[i].Lname << endl;
        file << emp[i].job << endl;
        file << emp[i].phone << endl;
        file << emp[i].address << endl;
        file << emp[i].mail << endl;
        file << emp[i].age << endl;
        file << emp[i].ID << endl;
        file << emp[i].payrate << endl;
        file << emp[i].workedhours << endl;
        file << emp[i].tax << endl;
        file << emp[i].gross << endl;
        file << emp[i].net << endl;
    }
    file.close();
}
```

There's a lot more to check go ahead and run the program ...

<i>Name</i>	<i>Code</i>
<i>Mohamed Ahmed Mohamed Rashad</i>	<i>20172098</i>
<i>Ahmed Rizk Ghallab</i>	<i>20172101</i>
<i>Ahmed Mohamed Ibrahim Abd El Rahim</i>	<i>20172013</i>
<i>Ahmed Gamal Mohamed Ahmed</i>	<i>20172062</i>