



Department of Computer Science

Maqsood Ahmed 38186

PF Assignment : 03

Question 01[CLO3: C3][20]

Part A

Write a program of currency converter using functions:

- 1. Press 1 :Call function which convert Pakistani Rs in to euro**
- 2. Press 2 :Call function which convert euro to yen**
- 3. Press 3 :Dollar in to euro**
- 4. Press 4 :Pakistani Rs to dollar**
- 5. Press 5 :Dollar in to yen**

Source Code:

```
#include <iostream>
using namespace std;

double rupeesToEuro(double currency) {
    return (currency * 0.0033);
}

double euroToYen(double currency) {
    return (currency * 161.4041);
}

double dollarToEuro(double currency) {
    return (currency * 1.0895);
}

double rupeesToDollar(double currency) {
    return (currency * 0.0036);
}

double dollarToYen(double currency) {
    return (currency * 148.1455);
}

void currencyConverterMenu() {
    int choice; double amount;
    do {
```

```

system("clear"); // clears the screen
cout << "\nPress 1 :Call function which convert Pakistani Rs in to euro\n"
    << "Press 2 :Call function which convert euro to yen\n"
    << "Press 3 :Dollar in to euro\n"
    << "Press 4 :Pakistani Rs to dollar\n"
    << "Press 5 :Dollar in to yen \n";
cout << ">> ";
cin >> choice;
} while(!(choice > 0 && choice < 6));

```

```

cout << "Enter amount: ";
cin >> amount;

```

```

switch (choice)
{

```

```

case 1:

```

```

    cout << "Converted amount is: " << rupeesToEuro(amount) << '\n';
    break;

```

```

case 2:

```

```

    cout << "Converted amount is: " << euroToYen(amount) << '\n';
    break;

```

```

case 3:

```

```

    cout << "Converted amount is: " << dollarToEuro(amount) << '\n';
    break;

```

```

case 4:

```

```

    cout << "Converted amount is: " << rupeesToDollar(amount) << '\n';
    break;

```

```

case 5:

```

```

    cout << "Converted amount is: " << dollarToYen(amount) << '\n';
    break;

```

```

default:

```

```

    break;

```

```

}

```

```

}

```

```

int main(void) {

    currencyConverterMenu();

    return EXIT_SUCCESS;
}

```

OUTPUT:

The screenshot shows a Visual Studio Code editor with a C++ file named `assign_3q1a.cpp` open. The code defines several conversion functions and a menu-driven interface. The terminal on the right shows the program's execution, where the user has selected option 1 to convert Pakistani Rupees to Euros. The input amount is 6842, and the output is 22.5786.

```

1 #include <iostream>
2 using namespace std;
3
4 double rupeesToEuro(double currency) {
5     return (currency * 0.0033);
6 }
7
8 double euroToYen(double currency) {
9     return (currency * 161.4041);
10 }
11
12 double dollarToEuro(double currency) {
13     return (currency * 1.0895);
14 }
15
16 double rupeesToDollar(double currency) {
17     return (currency * 0.0036);
18 }
19
20 double dollarToYen(double currency) {
21     return (currency * 148.1455);
22 }
23
24 void currencyConverterMenu() {
25     int choice; double amount;
26     do {
27         system("clear"); // clears the screen
28         cout << "\nPress 1 :Call function which convert Pakistani Rs in to eu
29         << "Press 2 :Call function which convert euro to yen\n"
30         << "Press 3 :Dollar in to euro\n"
31         << "Press 4 :Pakistani Rs to dollar\n"
32         << "Press 5 :Dollar in to yen \n";
33         cout << "> ";
34         cin >> choice;
35     } while(!(choice > 0 && choice < 6));
36
37     cout << "Enter amount: ";
38     cin >> amount;
39
40     switch (choice)
41     {
42     case 1:
43         cout << "Converted amount is: " << rupeesToEuro(amount) << "\n";

```

```

maqsood@maqsood-SBKPF: ~/My_Data/Code_Playground/Cpp/Cpp_Practice
Press 1 :Call function which convert Pakistani Rs in to euro
Press 2 :Call function which convert euro to yen
Press 3 :Dollar in to euro
Press 4 :Pakistani Rs to dollar
Press 5 :Dollar in to yen
>> 1
Enter amount: 6842
Converted amount is: 22.5786
maqsood@maqsood-SBKPF: ~/My_Data/Code_Playground/Cpp/Cpp_Practice$

```

Part B

Write a program which takes 10 student marks from user. The system shall allow user to use three functionalities i.e. insert (), delete(), update(). All the student marks are inserted in an array via functions.

Source Code:

```
#include <iostream>
using namespace std;

void insert(int arr[]) {

    int index;

    cout << "Enter the index where you want to insert(0-9): ";
    cin >> index;

    if(index >= 0 && index < 10) {
        cout << "What you want to insert: ";
        cin >> arr[index];
        cout << "Successfully inserted!\n";
    } else {
        cout << "Size exceeded! Try again\n";
        insert(arr);
    }
}

void update(int arr[]) {

    int index;

    cout << "Enter the index where you want to update(0-9): ";
    cin >> index;

    if(index >= 0 && index < 10) {
        cout << "What you want to update: ";
        cin >> arr[index];
        cout << "Successfully updated!\n";
    } else {
```

```

        cout << "Size exceeded! Try again\n";
        update(arr);
    }
}

```

```

void delete_(int arr[]) {

    int index;

    cout << "Enter the index where you want to delete(0-9): ";
    cin >> index;

    if(index >= 0 && index < 10) {
        cout << "Successfully deleted!\n";
        arr[index] = 0;
    } else {
        cout << "Size exceeded! Try again\n";
        delete_(arr);
    }
}

```

```

int main(void) {

    const int SIZE = 10; // constant size
    int array[SIZE];

    int choice;

    do {
        system("clear"); // clears the screen
        cout << "\nPress 1 :to insert\n"
              << "Press 2 :to update\n"
              << "Press 3 :to delete\n";
        cout << ">> ";
        cin >> choice;
    } while(!(choice > 0 && choice < 3));

    switch (choice)
    {
    case 1:
        insert(array);
        break;

    case 2:
        update(array);
        break;
    }
}

```

case 3:

```
delete_(array);  
break;
```

default:

```
break;  
}
```

```
return EXIT_SUCCESS;
```

```
}
```

OUTPUT:

The screenshot displays a Visual Studio Code window with a C++ file named `assign_3q1b.cpp` open. The code implements three functions: `insert`, `update`, and `delete`, each taking an array of integers and an index. The `insert` function prompts the user for an index (0-9) and a value to insert. The `update` function prompts for an index (0-9) and a new value to update. The `delete` function prompts for an index (0-9) to be removed. The terminal output shows the program being compiled and executed, with the user interacting with the prompts. The output shows successful insertion, update, and deletion operations.

```
assign_3q1b.cpp - Cpp_Practice - Visual Studio Code  
Jan 21 11:30 PM  
maqsood@maqsood-SBKPF: ~/My_Data/Code_Playground/Cpp/Cpp_Practice  
maqsood@maqsood-SBKPF: ~/My_Data/Code_Playground/Cpp/Cpp_Practice$ g++ assign_3q1b.cpp  
maqsood@maqsood-SBKPF: ~/My_Data/Code_Playground/Cpp/Cpp_Practice$ ./a.out  
Press 1 :to insert  
Press 2 :to update  
Press 3 :to delete  
>> 1  
Enter the index where you want to insert(0-9): 0  
What you want to insert: 30  
Successfully inserted!  
maqsood@maqsood-SBKPF: ~/My_Data/Code_Playground/Cpp/Cpp_Practice$ ./a.out  
Press 1 :to insert  
Press 2 :to update  
Press 3 :to delete  
>> 2  
Enter the index where you want to update(0-9): 0  
What you want to update: 89  
Successfully updated!  
maqsood@maqsood-SBKPF: ~/My_Data/Code_Playground/Cpp/Cpp_Practice$ ./a.out  
Press 1 :to insert  
Press 2 :to update  
Press 3 :to delete  
>> 3  
Enter the index where you want to delete(0-9): 0  
Successfully deleted!  
maqsood@maqsood-SBKPF: ~/My_Data/Code_Playground/Cpp/Cpp_Practice$
```

Question 02[CLO3: C3][20]

Part A

Write a C++ program that creates a structure called course that contains two members

- ➔ Name (string)
- ➔ Credit hr(int)

After that creates a structure called student that contains the following data members.

- ➔ Name(string)
- ➔ Age(int)
- ➔ Cms(int)
- ➔ Semester(int)
- ➔ Course (course)

Take the data member's value from the user and display the record of the 3 students.

Source Code:

```
#include <iostream>
using namespace std;

struct Course {
    string course_name;
    int credit_hr;
};

struct Student {
    string name;
    int age;
    int cms;
    int semester;
    Course course;
};
```



```

// takes student record from user as input
void getStudentRecord(Student &student) {

    cout << "Enter your name:    ";
    cin >> student.name;
    cout << "Enter your age:      ";
    cin >> student.age;
    cout << "Enter your cms:       ";
    cin >> student.cms;
    cout << "Enter your semester:   ";
    cin >> student.semester;
    cout << "Enter your course name: ";
    cin >> student.course.course_name;
    cout << "Enter credit hours:    ";
    cin >> student.course.credit_hr;
}

// displays student record
void displayStudentRecord(Student &student) {
    cout << "\n-----\n";
    cout << "Student name:    " << student.name << '\n';
    cout << "Student age:      " << student.age << '\n';
    cout << "Student cms:      " << student.cms << '\n';
    cout << "Enrolled semester: " << student.semester << '\n';
    cout << "Course name:      " << student.course.course_name << '\n';
    cout << "Student Name:     " << student.course.credit_hr << '\n';
}

int main(void) {

    system("clear"); // clears the screen

    Student student[3];
    for(int i = 0; i < 3; i++) {
        cout << "\n-----\n";
        cout << "\t\t STUDENT: " << i + 1 << "\n\n";
        getStudentRecord(student[i]);
    }

    system("clear"); // clears the screen

```

```

for(int i = 0; i < 3; i++) {
    cout << "\t\t STUDENT: " << i + 1 << "\n\n";
    displayStudentRecord(student[i]);
}

return EXIT_SUCCESS;
}

```

OUTPUT:

```

assign_3q2a.cpp - Cpp_Practice - Visual Studio Code
File Edit Selection View Go Run Terminal Help
C assign_3q1a.cpp assign_3q1b.cpp assign_3q2a.cpp X assign_3q2b.cpp
1 #include <iostream>
2 using namespace std;
3
4 struct Course {
5     string course_name;
6     int credit_hr;
7 };
8
9 struct Student {
10     string name;
11     int age;
12     int cms;
13     int semester;
14     Course course;
15 };
16
17 // takes student record from user as input
18 void getStudentRecord(Student &student) {
19
20     cout << "Enter your name: ";
21     cin >> student.name;
22     cout << "Enter your age: ";
23     cin >> student.age;
24     cout << "Enter your cms: ";
25     cin >> student.cms;
26     cout << "Enter your semester: ";
27     cin >> student.semester;
28     cout << "Enter your course name: ";
29     cin >> student.course.course_name;
30     cout << "Enter credit hours: ";
31     cin >> student.course.credit_hr;
32 }
33
34 // displays student record
35 void displayStudentRecord(Student &student) {
36     cout << "\n-----\n";
37     cout << "Student name: " << student.name << '\n';
38     cout << "Student age: " << student.age << '\n';
39     cout << "Student cms: " << student.cms << '\n';
40     cout << "Enrolled semester: " << student.semester << '\n';
41     cout << "Course name: " << student.course.course_name << '\n';
42     cout << "Student Name: " << student.course.credit_hr << '\n';
43 }

```

```

maqsood@maqsood-SBKPF: ~/My_Data/Code_Playground/Cpp/Cpp_Practice
STUDENT: 1
-----
Student name: Maqsood
Student age: 20
Student cms: 3
Enrolled semester: 2
Course name: Programming
Student Name: 4
STUDENT: 2
-----
Student name: Hanza
Student age: 20
Student cms: 4
Enrolled semester: 2
Course name: DLD
Student Name: 4
STUDENT: 3
-----
Student name: Zeeshan
Student age: 22
Student cms: 3
Enrolled semester: 2
Course name: DSA
Student Name: 4
maqsood@maqsood-SBKPF: ~/My_Data/Code_Playground/Cpp/Cpp_Practice$

```

Part B

Write a C++ program that creates a structure to manage a record of employee.

Employee

has 4 attribute.

- ➔ **Employee id**
- ➔ **Employee Name**
- ➔ **Employee age**
- ➔ **Employee salary**

Make a structure of employee having provided attributes and store 10 employee data and display it. Also display the name of those employee whose salary is greater than 5000.

Source Code:

```
#include <iostream>
using namespace std;

struct Employee {
    int employee_id;
    string employee_name;
    int employee_age;
    int employee_salary;
};

// takes student record from user as input
void getEmployeeRecord(Employee &employee) {

    cout << "Enter your name:    ";
    cin >> employee.employee_name;
    cout << "Enter your id:      ";
    cin >> employee.employee_id;
    cout << "Enter your age:         ";
    cin >> employee.employee_age;
    cout << "Enter your salary:    ";
    cin >> employee.employee_salary;

}

// displays student record
```

```

void displayEmployeeRecord(Employee &employee) {

    cout << "\n-----\n";
    cout << "Employee name:      " << employee.employee_name << "\n";
    cout << "Employee id:         " << employee.employee_id << "\n";
    cout << "Employee age:        " << employee.employee_age << "\n";
    cout << "Employee salary:     " << employee.employee_salary << "\n";
}

// displays record of employee whose salary is greater the 5000
void displayEmployeeNameSalaryGreaterThan5k(Employee &employee) {

    if(employee.employee_salary >= 5000) {
        cout << "Employee name: " << employee.employee_name << "\n";
    }
}

int main(void) {
    Employee employee[10];

    for(int i = 0; i < 10; i++) {
        cout << "\n-----\n";
        cout << "\t\t EMPLOYEE: " << i + 1 << "\n\n";
        getEmployeeRecord(employee[i]);
    }

    system("clear"); // clears the screen

    for(int i = 0; i < 10; i++) {
        cout << "\t\t EMPLOYEE: " << i + 1 << "\n\n";
        displayEmployeeRecord(employee[i]);
    }

    cout << "Displaying Employee name's whose salary is greater than 5k\n";
    for(int i = 0; i < 10; i++) {
        displayEmployeeNameSalaryGreaterThan5k(employee[i]);
    }
    return EXIT_SUCCESS;
}

```

OUTPUT:

```
assign_3q2b.cpp - Cpp_Practice - Visual Studio Code
Jan 21 11:35 PM
maqsood@maqsood-SBKPF: ~/My_Data/Code_Playground/Cpp/Cpp_Practice

1 #include <iostream>
2 using namespace std;
3
4 struct Employee {
5     int employee_id;
6     string employee_name;
7     int employee_age;
8     int employee_salary;
9 };
10
11 // takes student record from user as input
12 void getEmployeeRecord(Employee &employee) {
13
14     cout << "Enter your name: ";
15     cin >> employee.employee_name;
16     cout << "Enter your id: ";
17     cin >> employee.employee_id;
18     cout << "Enter your age: ";
19     cin >> employee.employee_age;
20     cout << "Enter your salary: ";
21     cin >> employee.employee_salary;
22 }
23
24
25 // displays student record
26 void displayEmployeeRecord(Employee &employee) {
27
28     cout << "\n-----\n";
29     cout << "Employee name: " << employee.employee_name << '\n';
30     cout << "Employee id: " << employee.employee_id << '\n';
31     cout << "Employee age: " << employee.employee_age << '\n';
32     cout << "Employee salary: " << employee.employee_salary << '\n';
33 }
34
35 // displays record of employee whose salary is greater the 5000
36 void displayEmployeeNameSalaryGreaterThen5k(Employee &employee) {
37
38     if(employee.employee_salary >= 5000) {
39         cout << "Employee name: " << employee.employee_name << '\n';
40     }
41 }
42
43
```

```
Enter your age: 20
Enter your salary: 4000

-----
EMPLOYEE: 2

Enter your name: hanza
Enter your id: 42
Enter your age: 22
Enter your salary: 5200

-----
EMPLOYEE: 3

Enter your name: Noor
Enter your id: 23
Enter your age: 20
Enter your salary: 4200

-----
EMPLOYEE: 4

Enter your name: Zameer
Enter your id: 32
Enter your age: 20
Enter your salary: 150

-----
EMPLOYEE: 5

Enter your name: Sajjad
Enter your id: 234
Enter your age: 21
Enter your salary: 2423

-----
EMPLOYEE: 6

Enter your name: Saml
Enter your id: 232
Enter your age: 22
Enter your salary: 42

-----
EMPLOYEE: 7

Enter your name: 
```

```
assign_3q2b.cpp - Cpp_Practice - Visual Studio Code
Jan 21 11:37 PM
maqsood@maqsood-SBKPF: ~/My_Data/Code_Playground/Cpp/Cpp_Practice

1 #include <iostream>
2 using namespace std;
3
4 struct Employee {
5     int employee_id;
6     string employee_name;
7     int employee_age;
8     int employee_salary;
9 };
10
11 // takes student record from user as input
12 void getEmployeeRecord(Employee &employee) {
13
14     cout << "Enter your name: ";
15     cin >> employee.employee_name;
16     cout << "Enter your id: ";
17     cin >> employee.employee_id;
18     cout << "Enter your age: ";
19     cin >> employee.employee_age;
20     cout << "Enter your salary: ";
21     cin >> employee.employee_salary;
22 }
23
24
25 // displays student record
26 void displayEmployeeRecord(Employee &employee) {
27
28     cout << "\n-----\n";
29     cout << "Employee name: " << employee.employee_name << '\n';
30     cout << "Employee id: " << employee.employee_id << '\n';
31     cout << "Employee age: " << employee.employee_age << '\n';
32     cout << "Employee salary: " << employee.employee_salary << '\n';
33 }
34
35 // displays record of employee whose salary is greater the 5000
36 void displayEmployeeNameSalaryGreaterThen5k(Employee &employee) {
37
38     if(employee.employee_salary >= 5000) {
39         cout << "Employee name: " << employee.employee_name << '\n';
40     }
41 }
42
43
```

```
Employee age: 21
Employee salary: 2423
EMPLOYEE: 6

-----
Employee name: Saml
Employee id: 232
Employee age: 22
Employee salary: 42
EMPLOYEE: 7

-----
Employee name: Zeeshan
Employee id: 234
Employee age: 22
Employee salary: 5200
EMPLOYEE: 8

-----
Employee name: Sajjad
Employee id: 4234
Employee age: 23
Employee salary: 234
EMPLOYEE: 9

-----
Employee name: Arslan
Employee id: 324
Employee age: 22
Employee salary: 52300
EMPLOYEE: 10

-----
Employee name: Raheen
Employee id: 42
Employee age: 22
Employee salary: 4238
Displaying Employee name's whose salary is greater than 5k
Employee name: hanza
Employee name: Zeeshan
Employee name: Arslan
maqsood@maqsood-SBKPF: ~/My_Data/Code_Playground/Cpp/Cpp_Practice$
```