National University
of Computer & Emerging Sciences

Program: BS (CS/SE)
Semester: Fall 2022
Course: CS1004 - Object Oriented Programming

Assignment#01
Marks: 60
Due Date: 19-09-2022

NOTE: Attempt all questions.

1. Consider the definition of the function main. [20]

```
int main()
{
        int x, y;
        char z;
        double rate, hours;
        double amount;
        .
        .
        .
}
```

The variables x, y, z, rate, and hours referred to in items a through f below are the variables of the function main. Each of the functions described must have the appropriate parameters to access these variables. Write the following definitions:

a. Write the definition of the function *initialize* that initializes x and y to 0 and z to the blank *character*.

b. Write the definition of the function *getHoursRate* that prompts the user to input the hours worked and rate per hour to initialize the variables hours and *rate* of the function main.

c. Write the definition of the value-returning function *payCheck* that calculates and returns the amount to be paid to an employee based on the hours worked and rate per hour. The hours worked and rate per hour are stored in the variables hours and rate, respectively, of the function main. The formula for calculating the amount to be paid is as follows: For the first 40 hours, the rate is the given rate; for hours over 40, the rate is 1.5 times the given rate.

d. Write the definition of the function *printCheck* that prints the hours worked, rate per hour, and the salary.

e. Write the definition of the function *funcOne* that prompts the user to input a number. The function then changes the value of x by assigning the value of the expression 2 times the (old) value of x plus the value of y minus the value entered by the user.

f. Write the definition of the function *nextChar* that sets the value of z to the next character stored in z.

g. Write the definition of a function main that tests each of these functions.

2. Consider the following C++ code: [20]

```
#include<iostream>
#include<cmath>

using namespace std;

void func1()
{
        cout << "To run the program, enter 1." << endl;
        cout << "To exit the pogram, enter 99." << endl;
        cout << "Enter 1 or 99: ";
}

void func2(/*formal parameters*/)
{
        // Write the body of func2.
}

int main()
{
        int num1, num2;
        double num3;
        int choice;


        do
        {
                func1 ();
                cin >> choice;
                if (choice == 1)
                {
                        func2 (num1, num2, num3);
                        cout << num1 << num2 << num3 << endl;
                }
        }
        while (choice != 99);

        return 0;
}
```

The function *func2* has three parameters of type *int*, *int*, and *double*, say a, b,and c, respectively. Write the definition of *func2* so that its action is as follows:

a. Prompt the user to input two integers and store the numbers in a and b, respectively.

b. If both of the numbers are nonzero:

   i. If a >= b, the value assigned to c is a to the power b, that is, a^b.

   ii. If a < b, the value assigned to c is b to the power a, that is, b^a.

c. If a is nonzero and b is zero, the value assigned to c is the square root of the absolute value of a.

d. If b is nonzero and a is zero, the value assigned to c is the square root of the absolute value of b.

e. Otherwise, the value assigned to c is 0.

The values of a, b, and c are passed back to the calling environment. After completing the definition of the *func2*, test run your program.

Note: The cmath header file contains the functions for power, square root and absolute. Details of these functions are as follows:

- The function for power is given as: pow(a, b) (is equal to a^b). For example, pow(2, 3) equals 8.
- The function for square root is given as: sqrt(x) (is equal to $\sqrt{x}$). For example, sqrt(9) equals 3.
- The function for absolute is given as: abs(x) (is equal to |x|). For example, abs(-9) equals 9.

3. The statements in the following program are not in the correct order.Rearrange the statements so that the program outputs the total time an employee spent on the job each day. The program asks the user to enter the employee's name, the arrival time (arrival hour, arrival minute, AM or PM),and departure time (departure hour, departure minute, AM or PM). The program also allows the user to run the program as long as the user wishes.After rearranging the statements, your program must be properly indented. [20]

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

void timeOnJob(int arvHr, int arvMin, bool arvIsAM,
int depHr, int depMin, bool depIsAM)
{
        int arvTimeInMin;
        int depTimeInMin;
        int timeOnJobInMin;
        else if (arvIsAM == true && depIsAM == false)
        {
                arvTimeInMin = arvHr * 60 + arvMin;
                depTimeInMin = depHr * 60 + depMin;
                timeOnJobInMin = (720 - arvTimeInMin) + depTimeInMin;
                cout << "Time spent of job: "
                        << timeOnJobInMin / 60 << " hour(s) and "
                        << timeOnJobInMin % 60 << " minutes." << endl;
        }
        else if (arvTimeInMin <= depTimeInMin)
        {
                timeOnJobInMin = depTimeInMin - arvTimeInMin;
                cout << "Time spent of job: "
                        << timeOnJobInMin / 60 << " hour(s) and "
                        << timeOnJobInMin % 60 << " minutes." << endl;
        }
        else cout << "Invalid input." << endl;
        if ((arvIsAM == true && depIsAM == true) ||
        (arvIsAM == false && depIsAM == false))
                cout << "Invalid input." << endl;
}
```

```
int main()
{
        string employeeName;
        int arrivalHr;
        int departureHr;
        int departureMin;
        bool departureAM;
        char response;
        char isAM;
        cout << "This program calculates the total time spent by an
                employee on the job." << endl;
        cout << "To run the program, enter (y/Y): ";
        cin >> response;

        while (response == 'y' || response == 'Y')
        {
                cout << "Enter employee's name: ";
                getline(cin, employeeName);

                if (isAM == 'y' || isAM == 'Y')
                        arrivalAM = true;
                else
                        arrivalAM = false;

                cout << "Enter departure hour: ";
                cin >> departureHr;

                cout << "Enter departure minute: ";
                cin >> departureMin;

                cout << "Enter (y/Y) if departure is before 12:00PM: ";
                cin >> isAM;

                if (isAM == 'y' || isAM == 'Y')
                        departureAM = true;
                else
                        departureAM = false;

                cout << employeeName << endl;
                timeOnJob(arrivalHr, arrivalMin, arrivalAM,
                        departureHr, departureMin, departureAM);

                cout << "Enter arrival hour: ";
                cin >> arrivalHr;
                cout << "Enter arrival minute: ";
                cin >> arrivalMin;
                cout << "Enter (y/Y) if arrival is before 12:00PM: ";
                cin >> isAM;

                int arrivalMin;
                bool arrivalAM;

                cout << "Run program again (y/Y): ";
                cin >> response;
        }
}
```