```
/**
 *
 * Write a program that computes the tax and tip on a restaurant
bill for a patron.
 *
 * 1. The program should accept the tax and tip
 * both as percentages from the command line when running the
program.
 * Meaning as argument passed into the main().
 * gcc bill.c
 * ./a.out 7 3
 *
 * 2. Display the meal cost, tax amount, tip amount, and total bill
on the screen.
 *
 * 3. The meal cost should be randomly chosen between the following
four:
    Salad: $9.95
    Soup: $4.55
    Sandwich: $13.25
    Pizza: $22.35
 * 3a. randomly chosen number
 * 3b. display menu
 * 3c. calculate meal cost
 *
 *
 * @author: Suhuan Pan
 */

#include <stdio.h>
#include <time.h>
#include <stdlib.h>


struct Menu {
   char *food;
   double price;
};


/***** prototype *****/
int random_num ();
double declare();

/*
 * 1.
 */
int main(int argc, char *argv[])
{
    printf("--- Restaurant Bill Program ---\n");

 //    1. validate the # of argument supplied must be 3 (include
the .exe)
    if (argc == 3) {
```

```
        printf("You enter tax = %s%%, tip = %s%%.\n", argv[1],
argv[2]);
    }

    else if (argc > 3) {
        printf("Too many arguments supplied.\n");
        return 0;
    }

    else {
        printf("Not enough arguments supplied.\n");
        return 0;
    }


    printf("\n--- Menu ---\n");


    double price = declare();
    if (price == 0.0) {
        printf("Error, system exits...");
        EXIT_FAILURE;
    }


    double tips = (atof(argv[2]) / 100) * price;

    double tax = (atof(argv[1]) / 100) * price;

    printf("\nPrice = $%f, Tax = $%f, Tips = $%f.\n", price, tax,
tips);

    printf("Total payment: $%f.\n", (price+tax+tips));

    return 0;
}

// 3a. generate a number between 1 to 4
int random_num () {

    time_t  time1;

    srand((unsigned ) time((&time1)));

    int meal_choice = rand() % 4 + 1;

    printf("Meal Choice: %d.\n", meal_choice);

    return meal_choice;
}

/*
 * Salad: $9.95
    Soup: $4.55
```

```
    Sandwich: $13.25
    Pizza: $22.35
*/
double declare() {

    struct Menu meal1;
    meal1.food = "(1) Salad";
    meal1.price = 9.95;
    printf("%s: $%f\n", meal1.food, meal1.price);

    struct Menu meal2;
    meal2.food = "(2) Soup";
    meal2.price = 4.55;
    printf("%s: $%f\n", meal2.food, meal2.price);


    struct Menu meal3;
    meal3.food = "(3) Sandwich";
    meal3.price = 13.25;
    printf("%s: $%f\n", meal3.food, meal3.price);

    struct Menu meal4;
    meal4.food = "(4) Pizza";
    meal4.price = 22.35;
    printf("%s: $%f\n", meal4.food, meal4.price);

    printf("\nGenerate a number for meal option:\n");

    int option = random_num();

    switch (option) {
        case 1:
            return meal1.price;
        case 2:
            return meal2.price;
        case 3:
            return meal3.price;
        case 4:
            return meal4.price;
        default:
            return 0.0;
    }
}
```