

Last login: Fri Sep 28 11:57:55 on ttys000
Caijuns-MacBook-Air:~ fennec2000\$ ssh qin@linprog5.cs.fsu.edu
== AUTHORIZED USERS ONLY ==
You are attempting to log into a FSU Computer Science Department machine.
Please be advised by continuing that you agree to the terms of the
Computer Access and Usage Policy of the Department of Computer Science.
== AUTHORIZED USERS ONLY ==

qin@linprog5.cs.fsu.edu's password:
Last login: Fri Sep 28 11:58:07 2018 from 10.135.215.152

```
%%%%%%%%%  
%                               %  
%  DEPARTMENT OF COMPUTER SCIENCE  %  
%    Florida State University    %  
%                               %  
%                               %  
%%%%%%%%%
```

No mudding, IRC, or other games from here.
Please get a private sector account for non-CS activities.

See <http://www.cs.fsu.edu> for departmental information.
See <http://system.cs.fsu.edu> for Systems information.
Send email to help@cs.fsu.edu for help.

Attention!!

If you forward your CS email to your FSU mailbox, make sure your FSU email address is up to date or you may miss important emails.

If you are over your disk or file quota, please check your CS email for instruction on how to locate and remove files

To download Tectia, go to link <https://system.cs.fsu.edu/ssh-tectia/>.
If you are off FSU campus, use following ID to access the page:

User Name: sshcs
Password: letmedownloadit

For students: check your CS email at <http://webmail2.cs.fsu.edu>

```
qin@linprog5.cs.fsu.edu:~>ls -F  
grade.cpp    krusty.exe*  recitation2.cpp  recitation3.exe*  ticketsys.cpp  
grade.exe*   Mail/        recitation2.exe* test.cpp           ticketsys.exe*  
krusty.cpp   README*      recitation3.cpp  test.exe*  
qin@linprog5.cs.fsu.edu:~>rm test*  
qin@linprog5.cs.fsu.edu:~>~vastola/usub/submit1 krusty.cpp  
***** File submitted *****  
Here are the contents of your submitted file:  
*****  
/* Name: Caijun Qin  
Date: 09/24/2018
```

Section: 6

Assignment: 3

Due Date: 10/01/2018

About this project:

This program is an automated order processor for the Krusty Krab. The customer will be able to select any whole number quantities of each of four food options listed in the menu. The customer will be able to see a receipt of the order at the end.

Assumptions: User only enters real numbers.

All work below was performed by Caijun Qin */

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

```
/**FUNCTION DECLARATIONS**/
```

```
void menu();
```

```
double OrderTotal(int patty, int drink, int fries, int pizza);
```

```
int main(int argc, char const *argv[]){
```

```
    /**VARIABLE DECLARATIONS**/
```

```
    //food choice number in menus
```

```
    int menuChoice = 1;
```

```
    double integerCheck;
```

```
    //number of each food ordered
```

```
    int patty = 0;
```

```
    int fries = 0;
```

```
    int drink = 0;
```

```
    int pizza = 0;
```

```
    //placeholder for number of menu items when one item is currently being orde
red
```

```
    int numOrders = 0;
```

```
    /**WHAT ACTUALLY APPEARS ON THE SCREEN TO THE USER**/
```

```
    cout << "Welcome to the Krusty Krab! \n\n";
```

```
    while(menuChoice != 0){
```

```
        //displays the menu
```

```
        menu();
```

```
        cout << "\n\n";
```

```
        //prompts for orders
```

```
        cout << "What would you like? (Enter 0 to finish order!): ";
```

```
        cin >> integerCheck;
```

```
        //checks if input is an integer
```

```
        if(integerCheck == ceil(integerCheck) || integerCheck == floor(integerCh
eck)){
```

```
            menuChoice = round(integerCheck);
```

```

} else {
    cout << "That isn't on the menu! Try again. \n\n";
    continue;
}

//takes the order
switch(menuChoice){
    //number of krabby patties
    case 1:
        cout << "How many: ";
        cin >> numOrders;
        patty += numOrders;
        numOrders = 0;
        cout << "\n";
        while(patty != abs(patty)){
            patty = 0;
            cout << "You can't order a negative amount of food! \n";
            cout << "Try Again: ";
            cin >> numOrders;
            patty += numOrders;
            numOrders = 0;
        }
        break;
    //number of barnacle fries
    case 2:
        cout << "How many: ";
        cin >> numOrders;
        fries += numOrders;
        numOrders = 0;
        cout << "\n";
        while(fries != abs(fries)){
            fries = 0;
            cout << "You can't order a negative amount of food! \n";
            cout << "Try Again: ";
            cin >> numOrders;
            fries += numOrders;
            numOrders = 0;
        }
        break;
    //number of kelp shakes
    case 3:
        cout << "How many: ";
        cin >> numOrders;
        drink += numOrders;
        numOrders = 0;
        cout << "\n";
        while(drink != abs(drink)){
            drink = 0;
            cout << "You can't order a negative amount of food! \n";
            cout << "Try Again: ";
            cin >> numOrders;
            drink += numOrders;

```

```

        numOrders = 0;
    }
    break;
//number of krusty krab pizzas
case 4:
    cout << "How many: ";
    cin >> numOrders;
    pizza += numOrders;
    numOrders = 0;
    cout << "\n";
    while(pizza != abs(pizza)){
        pizza = 0;
        cout << "You can't order a negative amount of food! \n";
        cout << "Try Again: ";
        cin >> numOrders;
        pizza += numOrders;
        numOrders = 0;
    }
    break;
//checks if order is complete
case 0:
    break;
//indicates invalid number chosen for the menu
default:
    cout << "That isn't on the menu! Try again. \n\n";
}
}

```

```

//reviews what the customer has ordered under the condition that s/he
//actually ordered at least one menu item

```

```

double total = OrderTotal(patty, drink, fries, pizza);

```

```

if(total > 0){
    cout << "\n\n";
    cout << "Your order: \n";
    //lists number of krabby patties ordered, if any
    if(patty == 1){
        cout << "1 krabby patty. \n";
    } else if(patty > 1){
        cout << patty << " krabby patties. \n";
    }
}

```

```

//lists number of barnacle fries ordered, if any
if(fries == 1){
    cout << "1 barnacle fries. \n";
} else if(fries > 1){
    cout << fries << " barnacle fries. \n";
}

```

```

//lists number of kelp shakes ordered, if any
if(drink == 1){
    cout << "1 kelp shake. \n";
} else if(drink > 1){

```

```

        cout << fries << " kelp shakes. \n";
    }

    //lists number of krusty krab pizzas ordered, if any
    if(pizza == 1){
        cout << "1 krust krab pizza. \n";
    } else if(pizza > 1){
        cout << pizza << " krusty krab pizzas. \n";
    }
    cout << "\n";

    cout << fixed << showpoint << setprecision(2) <<"Your total is $" << total << "\n";
    cout << "Enjoy the food!" << "\n\n";
    } else if(total == 0){
        cout << "Guess you aren't hungry." << "\n\n";
    }

    return 0;
}

/****FUNCTION DEFINITIONS****/
void menu()
{
    cout << "----- Menu ----- \n";
    cout << "1. \tKrabby Patty ..... $3.50\n";
    cout << "2. \tBarnicle Fries .... $1.50\n";
    cout << "3. \tKelp Shake ..... $1.00\n";
    cout << "4. \tKrusty Krab Pizza . $5.00\n";
}

double OrderTotal(int patty, int drink, int fries, int pizza)
{
    double total = patty * 3.50 + drink * 1.00 + fries * 1.50 + pizza * 5.00;
    return total;
}

```

```

***** END FEEDBACK *****
*   PLEASE VIEW ABOVE TO VERIFY THE CONTENTS OF   *
*               YOUR SUBMITTED FILE               *
*****
qin@linprog5.cs.fsu.edu:~>

```