{ PF LAB };

NATIONAL UNIVERSITY
of Computer & Emerging Sciences

Course: CL1002 – Programming Fundamentals.

Instructor: Sir Muhammad Usman.

Submitted by: Muhammad Rehan

Roll no. 22P-9106

Class: BSE-1A (Fall 2022)

Assignment No 3

Date: December 11th, 2022.

Department of Computer Science

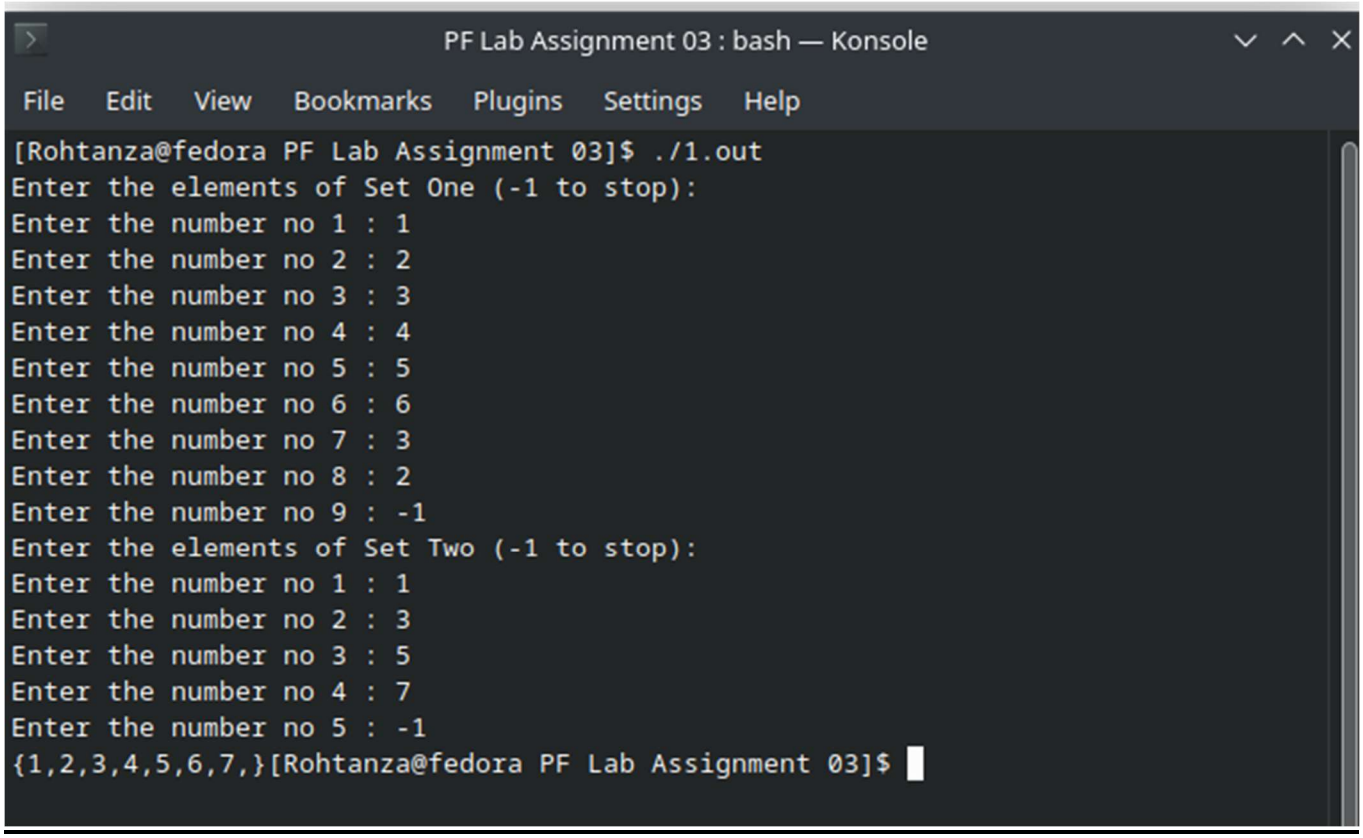
Problem 01:

Code:

```
#include <stdio.h>
#include <stdlib.h>
int *arraycreation();
void unionset(int *Setone, int *Settwo, int ArrayOnesize, int ArrayTwosize, int
union_size, int result[]);
int arraysize;
int main(void)
{
    printf("Enter the elements of Set One (-1 to stop):\n");
    int *Setone = arraycreation();
    int ArrayOnesize = arraysize;
    printf("Enter the elements of Set Two (-1 to stop):\n");
    int *Settwo = arraycreation();
    int ArrayTwosize = arraysize;
    int same = 0;
    for (int i = 0; i < ArrayOnesize; i++)
    {
        for (int j = 0; j < ArrayTwosize; j++)
            if (Setone[i] == Settwo[j])
                same++;
    }
    int union_size = (ArrayOnesize + ArrayTwosize) - same;
    int result[union_size];
    unionset(Setone, Settwo, ArrayOnesize, ArrayTwosize, union_size, result);
    return 0;
}
int *arraycreation()
{
    int *Numbers = NULL;
    size_t NumberOfAllocatedElements = 0;
    int TemporaryNumber;
    int j = 1;
    arraysize = 0;
    while (786)
    {
        printf("Enter the number no %d : ", j++);
        scanf("%d", &TemporaryNumber);
        if (TemporaryNumber == -1)
            break;
        int *NewNumbers = realloc(Numbers, ++NumberOfAllocatedElements * sizeof
*NewNumbers);
        if (!NewNumbers)
        {
            fprintf(stderr, "Error, unable to allocate memory.\n");
            exit(EXIT_FAILURE);
        }
        Numbers = NewNumbers;
        Numbers[NumberOfAllocatedElements - 1] = TemporaryNumber;
        arraysize++;
    }
```

```
    j = 0;
    return Numbers;
}
void unionset(int *Setone, int *Settwo, int ArrayOnesize, int ArrayTwosize, int
union_size, int result[])
{
    int i = 0, j = 0, x = 0, count = ArrayOnesize, flag = 1;

    for (i = 0; i < ArrayOnesize; i++)
    {
        result[i] = Setone[i];
    }

    for (i = 0; i < ArrayTwosize; i++)
    {
        flag = 1;
        x = 0;
        for (j = 0; flag == 1 && j < ArrayOnesize; j++)
        {
            if (Setone[j] == Settwo[i])
            {
                flag = 0;
            }
        }
        if (flag == 1)
        {
            result[count] = Settwo[i];
            count++;
        }
    }

    printf("Vettore generato:\n");
    for (i = 0; i < count; i++)
        printf(" %d ", result[i]);

    return ;
}
```

Screenshots:



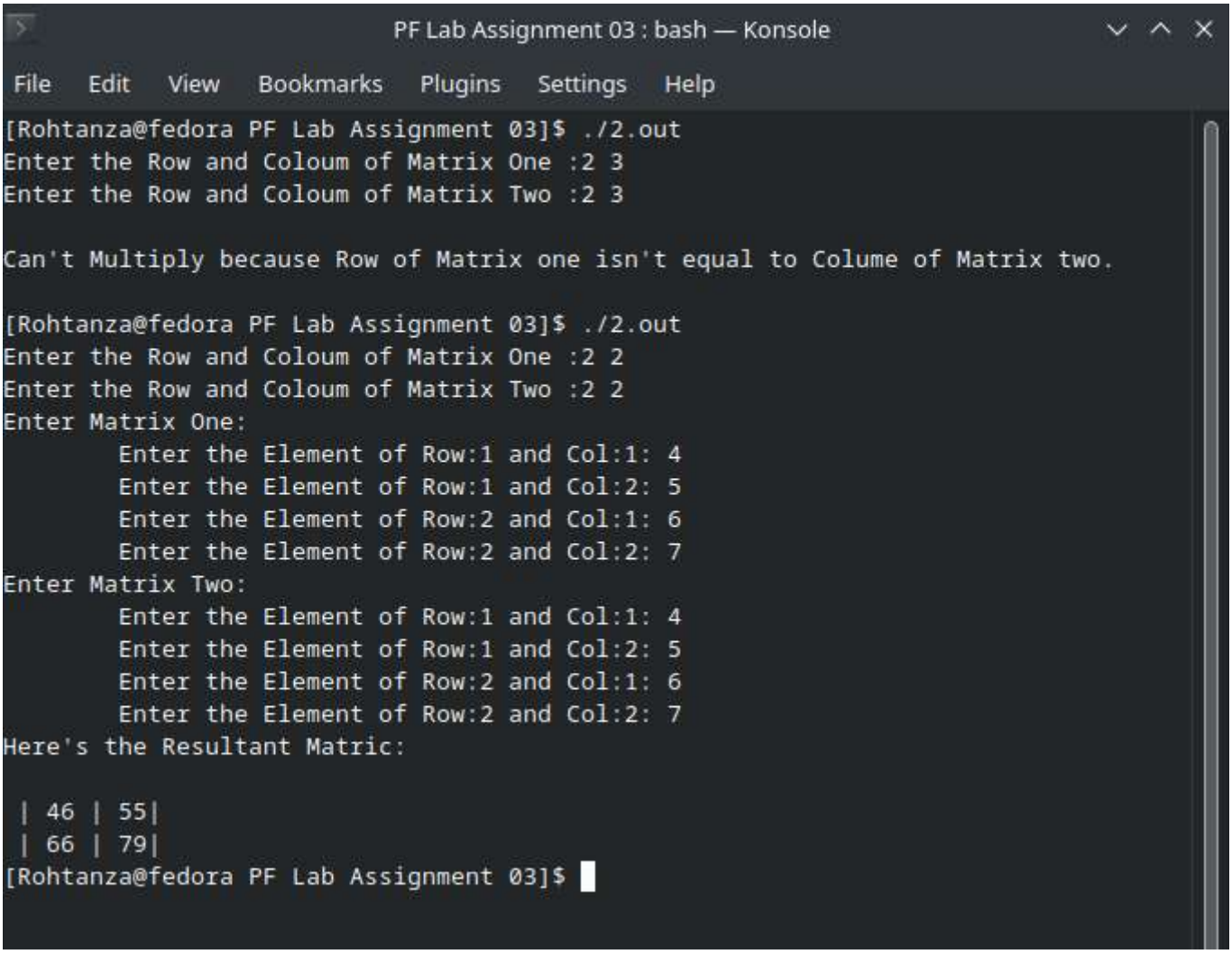
Problem 02:

Code:

```c
#include <stdio.h>
int main()
{
    int R_one, R_two, C_one, C_two;
    printf("Enter the Row and Coloum of Matrix One :");
    scanf("%d %d", &R_one, &C_one);
    printf("Enter the Row and Coloum of Matrix Two :");
    scanf("%d %d", &R_two, &C_two);
    if (C_one != R_two)
    {
        puts("\nCan't Multiply because Row of Matrix one isn't equal to Colume of Matrix two.\n");
        return 1;
    }
    int MatrixOne[R_one][C_one];
    int MatrixTwo[R_two][C_two];
    int Result[R_one][C_two];
    puts("Enter Matrix One:");
    for (int i = 0; i < R_one; i++)
    {
        for (int j = 0; j < C_one; j++)
        {
            printf("\tEnter the Element of Row:%d and Col:%d: ", (i + 1), (j + 1));
            scanf("%d", &MatrixOne[i][j]);
        }
    }
    puts("Enter Matrix Two:");
```

```
    for (int i = 0; i < R_two; i++)
    {
        for (int j = 0; j < C_two; j++)
        {
            printf("\tEnter the Element of Row:%d and Col:%d: ", (i + 1), (j + 1));
            scanf("%d", &MatrixTwo[i][j]);
        }
    }
    for (int i = 0; i < R_one; i++)
    {
        for (int j = 0; j < C_two; j++)
        {
            Result[i][j] = 0;
            for (int k = 0; k < C_one; k++)
            {
                Result[i][j] += MatrixOne[i][k] * MatrixTwo[k][j];
            }
        }
    }
    puts("Here's the Resultant Matric:\n");
    for (int i = 0; i < R_one; i++)
    {
        for (int j = 0; j < C_two; j++)
        {
            printf(" | %2d", Result[i][j]);
        }
        printf("| \n");
    }
    return 0;
}
```

Screenshot:



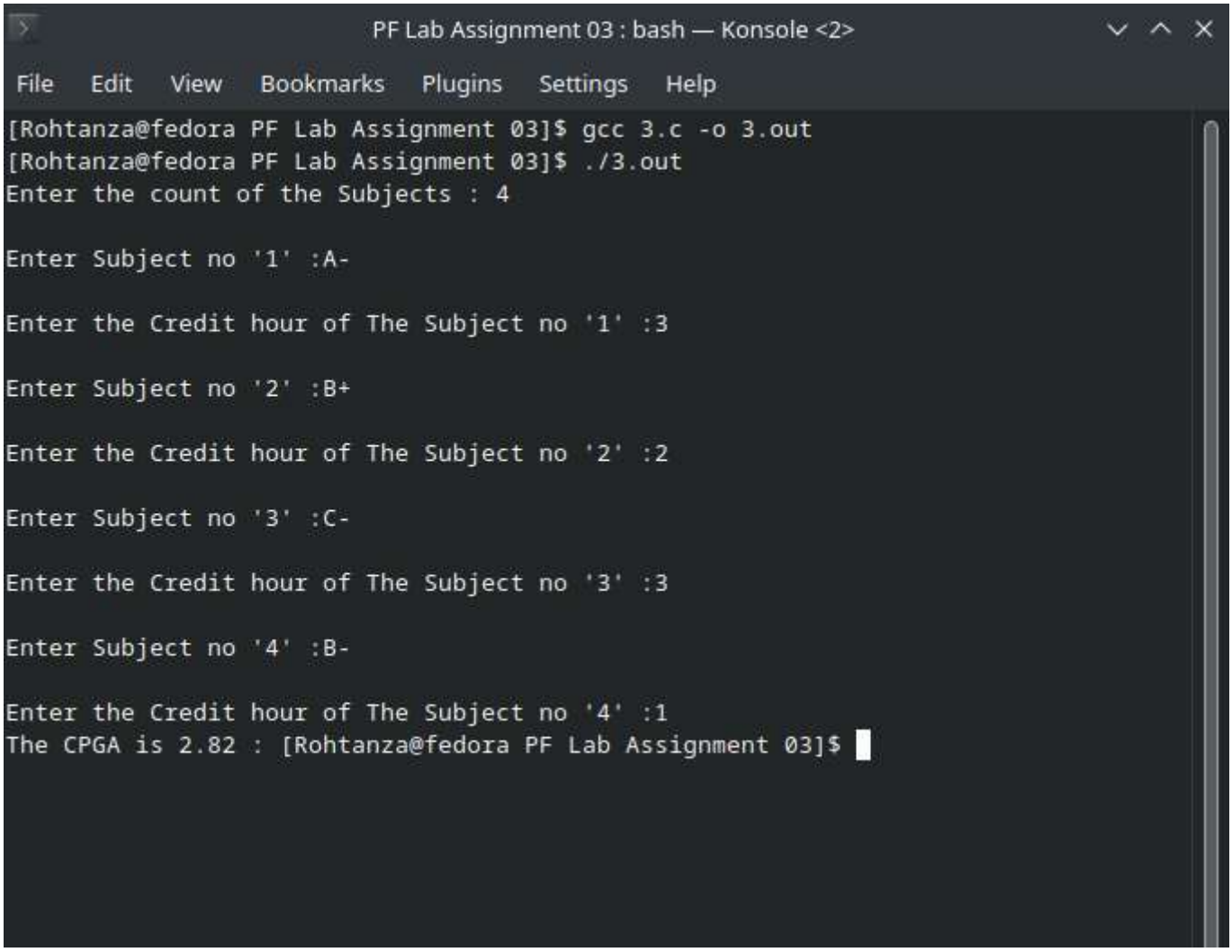
Problem 03:

Code:

```c
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
float CGPA(char **grades, float points[]);
int main()
{
    char *grades[] = {"A", "A-", "B+", "B", "B-", "C+", "C", "C-", "D+", "D", "F"};
    float points[] = {4.0, 3.67, 3.33, 3.0, 2.67, 2.33, 2.0, 1.67, 1.33, 1.0, 0};
    printf("The CPGA is %.2f : ", CGPA(grades, points));
    return 0;
}
float CGPA(char **grades, float points[])
{
    int Subject;
    float gradepoints, point,credit_hour_sum;
    credit_hour_sum = gradepoints = 0;
    printf("Enter the count of the Subjects : ");
    scanf("%d", &Subject);
    for (int i = 0; i < Subject; i++)
    {
        char grade[3];
```

```
        int j = 0;
        float credit_hour;
        printf("\nEnter Subject no \'%d\' :", (i + 1));
        scanf("%s", grade);
        printf("\nEnter the Credit hour of The Subject no \'%d\' :", (i + 1));
        scanf("%f", &credit_hour);
        while (1)
        {
            if (strcmp(grades[j], grade) == 0)
            {
                point = points[j];
                break;
            }
            j++;
            if (j > 11)
            {
                puts("Enter a valid Grade\nTry Again");
                exit(1);
            }
        }
        gradepoints = gradepoints + (point * credit_hour);
        credit_hour_sum += credit_hour;
    }
    return (gradepoints / credit_hour_sum);
}
```

Screenshot:



Problem 04:

Code:

```c
#include <stdio.h>
#include <stdlib.h>
#define Very_Vanilla_Chiller_Small 361
#define Very_Vanilla_Chiller_Regular 409
#define Cocoa_Loco_Small 361
#define Cocoa_Loco_Regular 409
#define CookiesCream_Small 361
#define CookiesCream_Regular 409
#define HazelNut_Small 369
#define HazelNut_Regular 461
#define Chocolate_Macadania_Small 369
#define Chocolate_Macadania_Regular 461
#define Italian_Small 369
#define Italian_Regular 461
#define Caramel_Small 369
#define Caramel_Regular 461
#define Tiramisu_Small 399
#define Tiramisu_Regular 509
#define Toffe_Small 399
```

```c
#define Toffe_Regular 509
#define Siganture_Small 300
#define Siganture_Regular 374
#define Mocha_Small 300
#define Mocha_Regular 361
#define IcedCaramel_Small 378
#define IcedCaramel_Regular 430
#define Americano_Small 252
#define Americano_Regular 274
#define Blueberry_Small 250
#define Blueberry_Regular 291
#define Lychee_Small 250
#define Lychee_Regular 291
#define GreenApple_Small 250
#define GreenApple_Regular 291
#define Peach_Small 250
#define Peach_Regular 291
#define AppleSoda_Small 335
#define AppleSoda_Regular 348
#define Lime_Small 335
#define Lime_Regular 361
#define Peachtea_Small 239
#define Peachtea_Regular 291
#define Lemontea_Small 239
#define Lemontea_Regular 291
#define Lycheetea_Small 239
#define Lycheetea_Regular 291
#define IcedChoco_Small 348
#define IcedChoco_Regular 365
#define WhiteChoco_Small 348
#define WhiteChoco_Regular 365
#define Delight_Small 348
#define Delight_Regular 400
#define IcedLime_Small 335
#define IcedLime_Regular 365
#define AppleChiller_Small 335
#define AppleChiller_Regular 365
#define ChaiChiller_Small 348
#define ChaiChiller_Regular 400
#define GreenTeaChiiler_Small 348
#define GreenTeaChiiler_Regular 400
void Lines(void);
int Type(int Budget);
int flavour4Expresso(int Budget);
int flavour4OverIce(int Budget);
int flavour4Cholocate(int Budget);
int flavour4Fusion(int Budget);
int Bill(int Order);
int Budget;
int main()
{
    int Choice;
    Lines();
    // Showing a Greeting Message.
```

```
    printf("\nHi There, Welcome to the Glouria Jeans.\n\nKindly Enter your Budget:");
    scanf("%d", &Budget);
    if (Budget < 239)
    {
        // Checking if budget is less than Budget.
        printf("\nSorry our least expensive Product costs \"239\"\nYou'r just \"%d\" buck
short.\n", 200 - Budget);
        Lines();
        printf("\n");
        return 0;
    }
    // Menu();
    Type(Budget);
    return 0;
}
int Type(int Budget)
{
    Lines();
    int Type;
    printf("\n\nEnter:\n\"1\" for Expresso & Mocha Chillers\n\"2\" for Over Ice\n\"3\" for
Cholocalate Chillers\n\"4\" for Fusion\n=>");
    scanf("%d", &Type);
    if (Type == -1)
    {
        // invalid Entry checker
        printf("-1");
        exit(0);
    }
    switch (Type)
    {
        // Giving the user an option to select either cup or cone to order
    case 1:
        flavour4Expresso(Budget);
        break;
    case 2:
        flavour4OverIce(Budget);
        break;
    case 3:
        flavour4Cholocate(Budget);
        break;
    case 4:
        flavour4Fusion(Budget);
        break;
    case -1:
        // invalid Entry checker
        printf("-1");
        break;

    default:
        break;
    }
}
int flavour4Expresso(int Budget)
{
```

```
    Lines();
    int flavour;
    int Size;
    printf("\nEnter the Flavour you like :):\n*1 for Very Vanilla Chiller\n*2 for Cocoa
Loco\n*3 for Cookies N' Cream\n*4 for Hazaelnut Mocha Chiller\n*5 for Chocolate Macadamia
Chillar\n*6 for Italian Cholio Chiller\n*7 for Caramel Nut Chiller\n*8 for Tiramisu
Chiller\n*9 for Toffe Nut Chiiler\n=>");
    scanf("%d", &flavour);
    if (flavour == -1)
    {
        // invalid Entry checker
        printf("-1");
        exit(0);
    }
    switch (flavour)
    {
        // Giving users an option to select flavor to order.
    case 1:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", Very_Vanilla_Chiller_Small, Very_Vanilla_Chiller_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(Very_Vanilla_Chiller_Small);
            break;
        case 2:
            Bill(Very_Vanilla_Chiller_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 2:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", Cocoa_Loco_Small, Cocoa_Loco_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(Cocoa_Loco_Small);
            break;
        case 2:
            Bill(Cocoa_Loco_Regular);
            break;
```

```
        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 3:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", CookiesCream_Small, CookiesCream_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(CookiesCream_Small);
            break;
        case 2:
            Bill(CookiesCream_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 4:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", HazelNut_Small, HazelNut_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(HazelNut_Small);
            break;
        case 2:
            Bill(HazelNut_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
```

```
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 5:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", Chocolate_Macadania_Small, Chocolate_Macadania_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(Chocolate_Macadania_Small);
            break;
        case 2:
            Bill(Chocolate_Macadania_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 6:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", Italian_Small, Italian_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(Italian_Small);
            break;
        case 2:
            Bill(Italian_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 7:
```

```
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", Caramel_Small, Caramel_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(Caramel_Small);
            break;
        case 2:
            Bill(Caramel_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 8:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", Tiramisu_Small, Tiramisu_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(Tiramisu_Small);
            break;
        case 2:
            Bill(Tiramisu_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 9:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", Toffe_Small, Toffe_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
```

```
        case 1:
            Bill(Toffe_Small);
            break;
        case 2:
            Bill(Toffe_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;

    default:
    Lines();
        printf("\nAt least read it carefully don't enter useless entries try again.\nthis
time read everything.\n");
        // Remaining User to read the program carefully,
        // Calling the function again.
        Type(Budget);
        break;
    }
}
int flavour4OverIce(int Budget)
{
    Lines();
    int flavour;
    int Size;
    printf("\nEnter the Flavour you like :):\n*1 for Signature Iced Coffe\n*2 for Iced
Mocha\n*3 for Iced Caramel Latte\n*4 for Iced Americano\n*5 for BlueBerry Lemonade\n*6 for
Lychee Lemonade\n*7 for Green Apple Lemonade\n*8 for Peach Lemonade\n*9 for Apple
Soda\n*10 for Lime Soda\n*11 for Ice Teas\n=>");
    scanf("%d", &flavour);
    if (flavour == -1)
    {
        // invalid Entry checker
        printf("-1");
        exit(0);
    }
    switch (flavour)
    {
        // Giving users an option to select flavor to order.
    case 1:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", Siganture_Small, Siganture_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
```

```
        case 1:
            Bill(Siganture_Small);
            break;
        case 2:
            Bill(Siganture_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 2:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", Mocha_Small, Mocha_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(Mocha_Small);
            break;
        case 2:
            Bill(Mocha_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 3:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", IcedCaramel_Small, IcedCaramel_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(IcedCaramel_Small);
            break;
        case 2:
            Bill(IcedCaramel_Regular);
            break;
```

```
        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 4:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", Americano_Small, Americano_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(Americano_Small);
            break;
        case 2:
            Bill(Americano_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 5:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", Blueberry_Small, Blueberry_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(Blueberry_Small);
            break;
        case 2:
            Bill(Blueberry_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
```

```
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 6:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", Lychee_Small, Lychee_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(Lychee_Small);
            break;
        case 2:
            Bill(Lychee_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 7:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", GreenApple_Small, GreenApple_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(GreenApple_Small);
            break;
        case 2:
            Bill(GreenApple_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 8:
```

```
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", Peach_Small, Peach_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(Peach_Small);
            break;
        case 2:
            Bill(Peach_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 9:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", AppleSoda_Small, AppleSoda_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(AppleSoda_Small);
            break;
        case 2:
            Bill(AppleSoda_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 10:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", Lime_Small, Lime_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
```

```
        case 1:
            Bill(Lime_Small);
            break;
        case 2:
            Bill(Lime_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 11:
        int TeaType;
        printf("\nEnter the Tea you would like:\n*1 for Peach Tea for %d \n*2 for Lemon
Tea for %d\n*3 for Lychee Tea for %d\n=>");
        switch (TeaType)
        {
        case 1:
            Lines();
            printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular
for %d\n=>", Peachtea_Small, Peachtea_Regular);
            scanf("%d", &Size);
            switch (Size)
            {
            case 1:
                Bill(Peachtea_Small);
                break;
            case 2:
                Bill(Peachtea_Regular);
                break;

            default:
            Lines();
                printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
                // Remaining User to read the program carefully,
                // Calling the function again.
                flavour4Expresso(Budget);
                break;
            }
            break;
        case 2:
            Lines();
            printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular
for %d\n=>", Lemontea_Small, Lemontea_Regular);
            scanf("%d", &Size);
            switch (Size)
            {
```

```
            case 1:
                Bill(Lemontea_Small);
                break;
            case 2:
                Bill(Lemontea_Regular);
                break;

            default:
            Lines();
                printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
                // Remaining User to read the program carefully,
                // Calling the function again.
                flavour4Expresso(Budget);
                break;
            }
            break;
        case 3:
            Lines();
            printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular
for %d\n=>", Lycheetea_Small, Lycheetea_Regular);
            scanf("%d", &Size);
            switch (Size)
            {
            case 1:
                Bill(Lycheetea_Small);
                break;
            case 2:
                Bill(Lycheetea_Regular);
                break;

            default:
            Lines();
                printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
                // Remaining User to read the program carefully,
                // Calling the function again.
                flavour4Expresso(Budget);
                break;
            }
            break;
        }

    default:
    Lines();
        printf("\nAt least read it carefully don't enter useless entries try again.\nthis
time read everything.\n");
        // Remaining User to read the program carefully,
        // Calling the function again.
        Type(Budget);
        break;
    }
}
int flavour4Cholocate(int Budget)
```

```
{
    Lines();
    int flavour;
    int Size;
    printf("\nEnter the Flavour you like :):\n*1 for Orginal Iced Chiller\n*2 for White
Iced Chocolate\n*3 for Chocolate Delight\n=>");
    scanf("%d", &flavour);
    if (flavour == -1)
    {
        // invalid Entry checker
        printf("-1");
        exit(0);
    }
    switch (flavour)
    {
        // Giving users an option to select flavor to order.
    case 1:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", IcedChoco_Small, IcedChoco_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(IcedChoco_Small);
            break;
        case 2:
            Bill(IcedChoco_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 2:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", WhiteChoco_Small, WhiteChoco_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(WhiteChoco_Small);
            break;
        case 2:
            Bill(WhiteChoco_Regular);
            break;
```

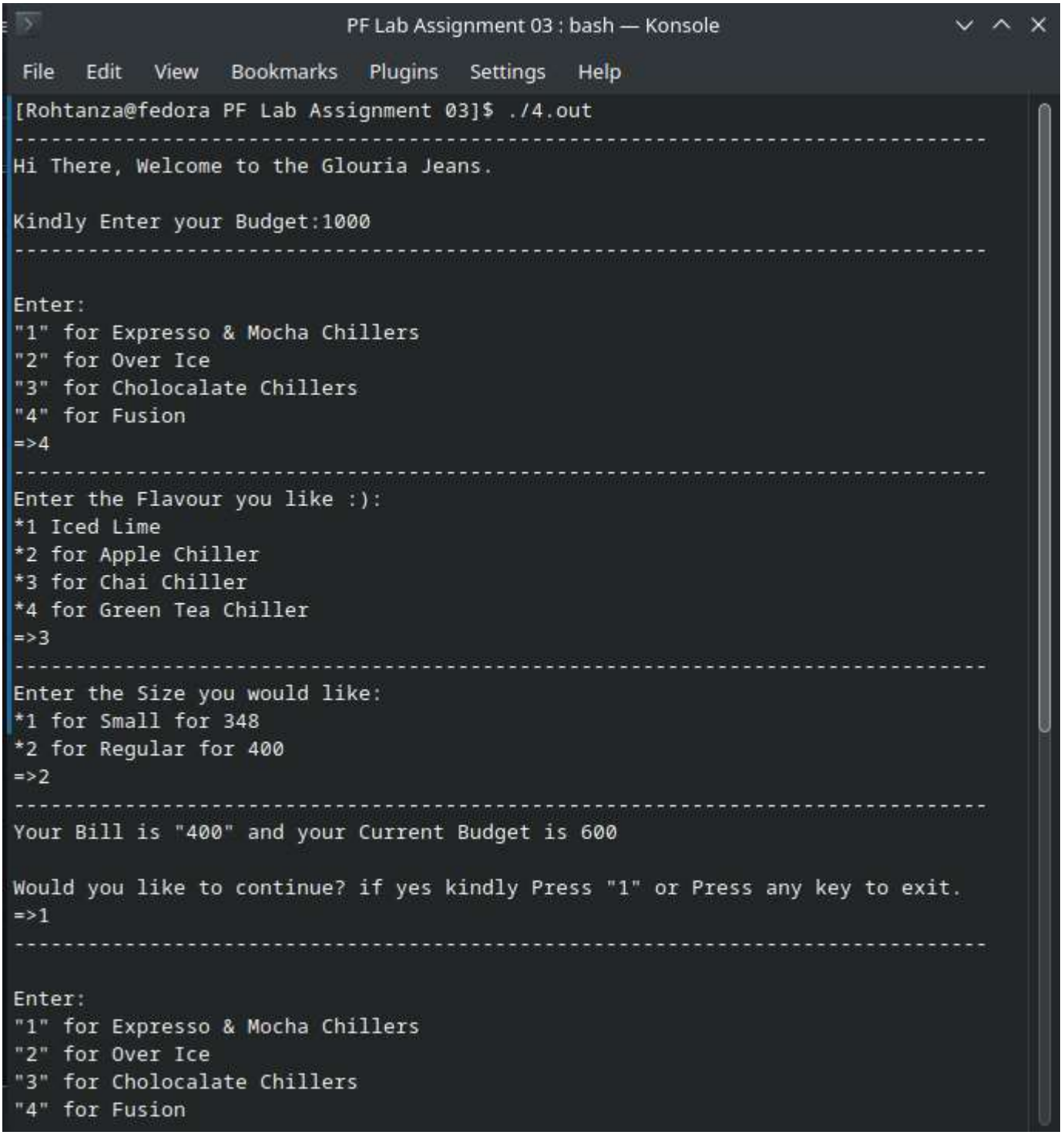
```
        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 3:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", Delight_Small, Delight_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(Delight_Small);
            break;
        case 2:
            Bill(Delight_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    default:
    Lines();
        printf("\nAt least read it carefully don't enter useless entries try again.\nthis
time read everything.\n");
        // Remaining User to read the program carefully,
        // Calling the function again.
        Type(Budget);
        break;
    }
}
int flavour4Fusion(int Budget)
{
    Lines();
    int flavour;
    int Size;
    printf("\nEnter the Flavour you like :):\n*1 Iced Lime\n*2 for Apple Chiller\n*3 for
Chai Chiller\n*4 for Green Tea Chiller\n=>");
    scanf("%d", &flavour);
    if (flavour == -1)
    {
```

```
        // invalid Entry checker
        printf("-1");
        exit(0);
    }
    switch (flavour)
    {
        // Giving users an option to select flavor to order.
    case 1:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", IcedLime_Small, IcedLime_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(IcedLime_Small);
            break;
        case 2:
            Bill(IcedLime_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 2:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", AppleChiller_Small, AppleChiller_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(AppleChiller_Small);
            break;
        case 2:
            Bill(AppleChiller_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
```

```
        break;
    case 3:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", ChaiChiller_Small, ChaiChiller_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(ChaiChiller_Small);
            break;
        case 2:
            Bill(ChaiChiller_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    case 4:
        Lines();
        printf("\nEnter the Size you would like:\n*1 for Small for %d \n*2 for Regular for
%d\n=>", GreenTeaChiiler_Small, GreenTeaChiiler_Regular);
        scanf("%d", &Size);
        switch (Size)
        {
        case 1:
            Bill(GreenTeaChiiler_Small);
            break;
        case 2:
            Bill(GreenTeaChiiler_Regular);
            break;

        default:
        Lines();
            printf("\nAt least read it carefully don't enter useless entries try
again.\nthis time read everything.\n");
            // Remaining User to read the program carefully,
            // Calling the function again.
            flavour4Expresso(Budget);
            break;
        }
        break;
    default:
    Lines();
        printf("\nAt least read it carefully don't enter useless entries try again.\nthis
time read everything.\n");
        // Remaining User to read the program carefully,
```

```
        // Calling the function again.
        Type(Budget);
        break;
    }
}
void Lines(void)
{
    // Just a loop to create lines.
    for (int i = 1; i < 80; i++)
    {
        printf("-");
    }
}
int Bill(int Order)
{
    // This Function proceeds the bill with the selected item.
    int Countiue;
    Lines();
    printf("\nYour Bill is \"%d\" and your Current Budget is %d", Order, Budget - Order);
    // Updating the current budget.
    Budget = Budget - Order;
    // Asking if the user wants to Continue.
    printf("\n\nWould you like to continue? if yes kindly Press \"1\" or Press any key to
exit.\n=>");
    scanf("%d", &Countiue);
    if (Countiue == -1)
    {
        // invalid Entry checker
        printf("-1");
        exit(0);
    }
    if (Countiue == 1)
    {
        // if the player wants to continue then the player just has a budget of more than
239.
        if (Budget >= 239)
        {
            // If yes then proceed to order again.
            Type(Budget);
        }
        else
        {
            // If the Budget is less than 239 then excites the program.
            printf("\nSorry you are out of Budget :(\n\nHope to See you Soon.\n");
            exit(0);
        }
    }
    else
    {
        // If a player wants to leave the program then say the player the greetings.
        printf("\nThank you for Shopping here.\n\n See you soon :)\n");
        exit(0);
    }
}
```

Screenshot:


PF Lab Assignment 03 : bash — Konsole
File Edit View Bookmarks Plugins Settings Help
[Rohtanza@fedora PF Lab Assignment 03]$ ./4.out
Hi There, Welcome to the Glouria Jeans.
Kindly Enter your Budget:1000
Enter:
"1" for Expresso & Mocha Chillers
"2" for Over Ice
"3" for Cholocalate Chillers
"4" for Fusion
=>4
Enter the Flavour you like :):
*1 Iced Lime
*2 for Apple Chiller
*3 for Chai Chiller
*4 for Green Tea Chiller
=>3
Enter the Size you would like:
*1 for Small for 348
*2 for Regular for 400
=>2
Your Bill is "400" and your Current Budget is 600
Would you like to continue? if yes kindly Press "1" or Press any key to exit.
=>1
Enter:
"1" for Expresso & Mocha Chillers
"2" for Over Ice
"3" for Cholocalate Chillers
"4" for Fusion


PF Lab Assignment 03 : bash — Konsole
File Edit View Bookmarks Plugins Settings Help
*2 for Regular for 400
=>2
Your Bill is "400" and your Current Budget is 600
Would you like to continue? if yes kindly Press "1" or Press any key to exit.
=>1
Enter:
"1" for Expresso & Mocha Chillers
"2" for Over Ice
"3" for Cholocalate Chillers
"4" for Fusion
=>4
Enter the Flavour you like :):
*1 Iced Lime
*2 for Apple Chiller
*3 for Chai Chiller
*4 for Green Tea Chiller
=>2
Enter the Size you would like:
*1 for Small for 335
*2 for Regular for 365
=>2
Your Bill is "365" and your Current Budget is 235
Would you like to continue? if yes kindly Press "1" or Press any key to exit.
=>1
Sorry you are out of Budget :(
Hope to See you Soon.
[Rohtanza@fedora PF Lab Assignment 03]$