Introduction to Programming in C

Assignment 2

Contents

[Exercise 1: Implementation and Source Code 1](#_Toc26889161)

[Exercise 1: Specification 5](#_Toc26889162)

[Exercise 1: Analysis 5](#_Toc26889163)

[Exercise 1: Design 6](#_Toc26889164)

[Exercise 1: Testing and verification 7](#_Toc26889165)

[Exercise 2: Implementation and Source Code 8](#_Toc26889166)

[Exercise 2: Specification 11](#_Toc26889167)

[Exercise 2: Analysis 11](#_Toc26889168)

[Exercise 2: Testing and verification 13](#_Toc26889169)

# Exercise 1: Implementation and Source Code

#include <stdio.h>

#define TELEPHONE\_CHARGE 0.05

#define INTERNET\_CHARGE 0.02

#define TELEVISION\_BASIC 0.01

#define TELEVISION\_PREMIUM 0.02

/\*Functions\*/

void details(unsigned int \*account\_number, char \*code, unsigned int \*mins);

double telephone(unsigned int a);

double internet(unsigned int b);

double television(unsigned int c, char d);

char info(char get\_pack);

void print\_data(unsigned account\_number, double charge);

/\*Main Function\*/

void main(void)

{

/\*Variables\*/

unsigned int account\_number, mins; //Unsigned because it should not be a minus value

char code, pack;

double charge;

/\*Part 1: Calls function to get input from user\*/

details(&account\_number, &code, &mins);

/\*Part 2: Decides which function to run\*/

switch(code)

{

/\*Part 3: Runs to use function for telephone option\*/

case 'p':

case 'P':

charge = telephone(mins);

break;

/\*Part 4: Runs to use function for internet option\*/

case 'i':

case 'I':

charge = internet(mins);

break;

/\*Part 5: Runs to use function for television option\*/

case 't':

case 'T':

/\*Checks what pack if this option is chosen\*/

pack = info(pack);

charge = television(mins, pack);

break;

default:

printf("\nERROR: The option entered is invalid"); //Prints error message

}

/\*Part 6\*/

print\_data(account\_number, charge);

}

void details(unsigned int \*account\_number, char \*code, unsigned int \*mins)

{

/\*Get account number\*/

printf("Enter account number: ");

fflush(stdin);

scanf("%u", \*&account\_number);

/\*Get option\*/

printf("\nP: Telephone\nI: Internet\nT: Television\nEnter service code option: ");

fflush(stdin);

scanf("%c", \*&code);

/\*Get minutes\*/

printf("\nEnter how many minutes: ");

fflush(stdin);

scanf("%u", \*&mins);

}

/\*Function calculates charge for telephone\*/

double telephone(unsigned int a)

{

double due = 15.0;

due = due + (a \* TELEPHONE\_CHARGE);

return due;

}

/\*Function calculates charge for internet\*/

double internet(unsigned int b)

{

double due = 0.0;

if(b > 0);

{

due = 20.0;

}

if(b > 1000)

{

b = b - 1000;

due = due + (b \* INTERNET\_CHARGE);

}

return due;

}

/\*Function gets pack type for television\*/

char info(char get\_pack)

{

printf("\nPack types:\nB: Basic\nP: Premium\nEnter 'B' or 'P' to select pack: ");

fflush(stdin);

scanf("%c", &get\_pack);

return get\_pack;

}

/\*Calculates television charge\*/

double television(unsigned int c, char d)

{

double due = 0.0;

switch(d)

{

case 'b':

case 'B':

if(c > 0);

{

due = 5.0;

}

if(c > 60)

{

c = c - 60;

due = due + (c \* TELEVISION\_BASIC);

}

break;

case 'p':

case 'P':

if(c > 0);

{

due = 10.0;

}

if(c > 60)

{

c = c - 60;

due = due + (c \* TELEVISION\_PREMIUM);

}

break;

default:

printf("\nERROR: The pack entered is invalid");

}

return due;

}

/\*Function prints data\*/

void print\_data(unsigned int account\_number, double charge)

{

printf("\nAccount Number: %u", account\_number);

printf("\nAmmount Due: %.2f GBP", charge);

}

# Exercise 1: Specification

Write a program that calculates the total monthly bills for a service charge. The program allows the user to enter their account number. The program will the continue to allow the user to enter the single service that they use from three categories: telephone, internet, and television. If television is selected, then it will ask an additional question of what package that they have. Once this has been done, it will calculate the total charge. Finally, both the account number and the final charge will be output to the user.

# Exercise 1: Analysis

Inputs:

* account\_number – (unsigned integer) this is because it should always be a positive number
* code – (char) to select service option
* pack - (char) type of television pack
* mins – (unsigned int) How many minutes used

Outputs:

* account\_number – (unsigned integer) printed to user
* charge – (double) final calculation printed to the user

# Exercise 1: Design

Part 1:

* Runs function to get information from user
* Gets account number from user
* Gets service code from user
* Gets minutes from user
* All returned to main function by changing value at address

Part 2:

* Determines which function to run
* Switch determines if that option is selected

Part 3:

* Sets due amount to £15
* Increases amount due depending on how many minutes used
* Returns due

Part 4:

* Sets due to £0
* If more than 0 minutes are spent, instantly jumps to £20
* If more than 1000 minutes are spent, it removes 1000 minutes (because they are at no extra charge) and then calculates the additional price of minutes remaining.
* Returns final charge due

Part 5:

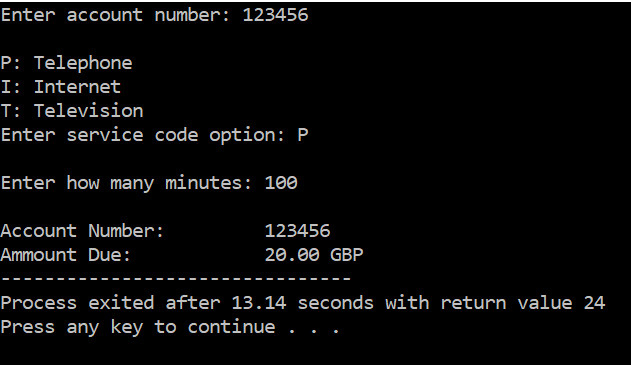
* Comprised of two functions
* First checks what package customer has
* Second calculates final price based on that.
* Sets due to £0.00
* Uses switch to determine package type
* If minutes are more than 0, adds the initial charge.
* If over the 60 minutes, removes 60 minutes and adds additional cost to what is due
* Returns to main function.

Part 6:

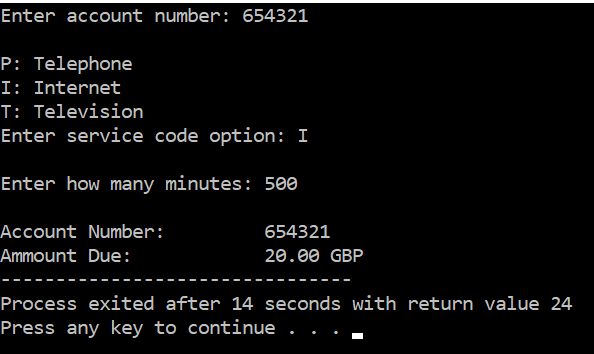
* Prints account number
* Prints final charge

# Exercise 1: Testing and verification

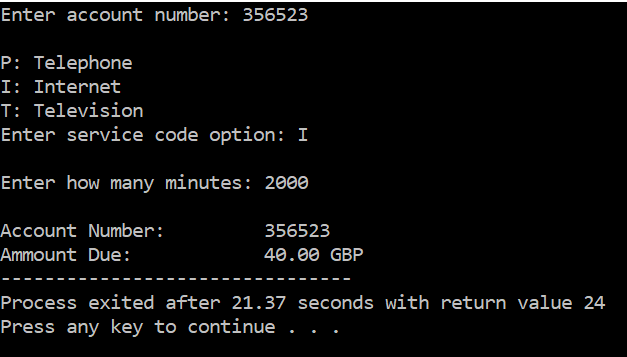
i)



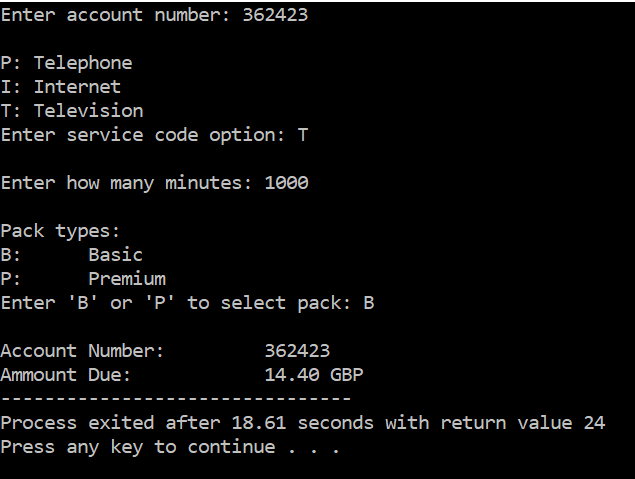
ii)



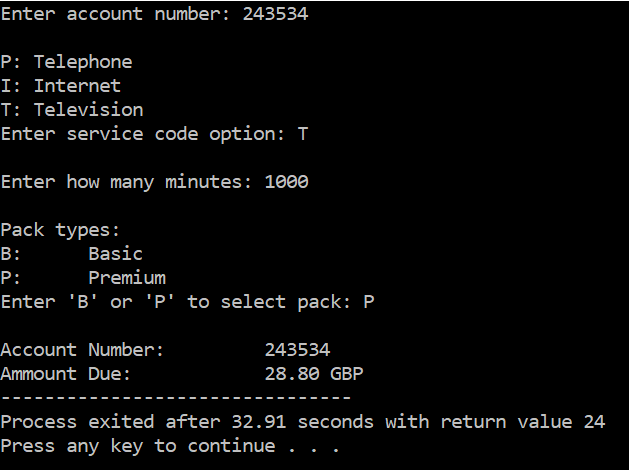
iii)



iv)



v)



It appears all test cases work correctly.

# Exercise 2: Implementation and Source Code

#include <stdio.h>

/\*Functions used in program\*/

float mean(float nums, int end);

float minus1(float currentnum, float avg\_x);

float kurtosis(float totalfinal, float last);

float calckurtosis(float totalfinal, float totalfinal2);

/\*Main function\*/

void main(void)

{

int i, last = 10, j;

float your\_nums[10];

float a, avg\_x, total = 0, currentnum, finx, final, final2, totalfinal = 0.0, totalfinal2 = 0, complete1, complete2, kurtosisfinal = 0.0;

/\*Part 1\*/

for(i=0; i < 10; i++)

{

printf("Enter number (or negative to end): ");

scanf("%f", &a);

if(a <= 0)

{

last = i;

i = 10;

}

fflush(stdin);

\*(your\_nums + i) = a;

}

/\*Part 2\*/

for(i=0; i < last; i++)

{

total = total + \*(your\_nums + i);

}

/\*Part 3\*/

for(i=0; i<last; i++)

{

avg\_x = mean(total, last);

}

/\*Part 4\*/

for(i=0; i<last; i++)

{

currentnum = \*(your\_nums+i);

finx = minus1(currentnum, avg\_x);

final = 0.0;

final2 = 0.0;

totalfinal = totalfinal + finx\*finx\*finx\*finx;

totalfinal2 = totalfinal2 + finx\*finx;

}

/\*part 5\*/

complete1 = kurtosis(totalfinal, last);

complete2 = kurtosis(totalfinal2, last);

/\*Part 6\*/

kurtosisfinal = calckurtosis(complete1, complete2);

/\*Part 7\*/

for(i=0; i<last; i++)

{

printf("\nThe value of %d = %f", i + 1, \*(your\_nums + i));

}

printf("\n-----------\nFinal result: %f", kurtosisfinal);

}

float mean(float nums, int end)

{

float result, div;

div = end;

result = 1 / div;

result = result \* nums;

return result;

}

float minus1(float currentnum, float avg\_x)

{

float result = 0;

result = currentnum - avg\_x;

return result;

}

float kurtosis(float totalfinal, float end)

{

float result, div;

div = end;

result = 1 / div;

result = result \* totalfinal;

return result;

}

float calckurtosis(float totalfinal, float totalfinal2)

{

float result, result1;

result1 = totalfinal2 \* totalfinal2;

result = totalfinal / result1;

return result;

}

# Exercise 2: Specification

Write a program that is able to calculate the kurtosis of up to 10 numbers that are entered. The program should stop accepting numbers if either the user enters 10 numbers or if they enter a negative value. If a negative value is entered, then the negative value itself should not be factored in. Both the values entered by the user and the final answer should be printed to the user.

# Exercise 2: Analysis

Inputs:

* Users numbers added to float array

Outputs:

* Users numbers
* Final answer

Exercise 2: Design

Part 1

* Uses for loop to 10
* Gets each number from the user
* Each number added to array
* If number entered is below 0, changes I to 10 immediately thus ending the loop.

Part 2

* Uses for loop to find the first part of the equation by adding the values
* For loop ends after final number entered is considered

Part 3

* For loop that continues until final number is used.
* Runs function to calculate mean
* Returns result to main function

Part 4

* Runs function to calculate next part of equation

Part 5

* Runs function twice.
* First it uses totalfinal and last to calculate part of the top part of the equation
* Next it uses totalfinal2 and last to calculate part of the bottom part of the equation

Part 6

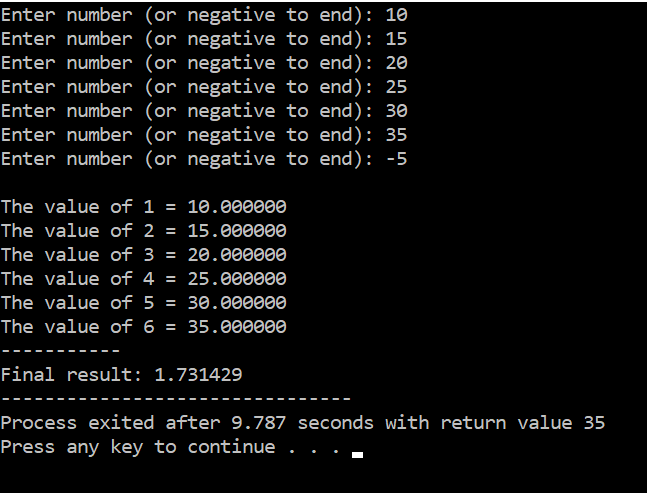
* Runs function to calculate kurtosis

Part 7

* Uses for loop to prints the input from user
* Prints final answer

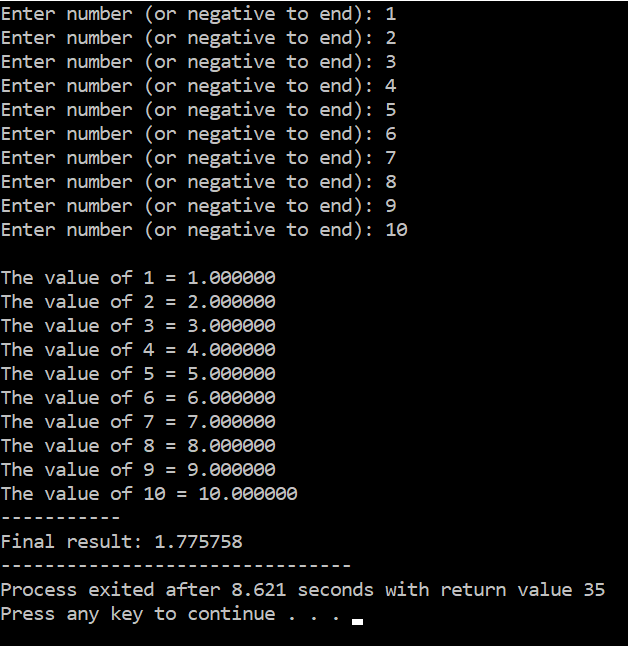
# Exercise 2: Testing and verification

i)



By calculating this manually, it has been determined that this seems to be what is expected and therefore the program appear to work correctly in this instance.

ii)



If you attempt to type in numbers from 1 – 15, you are unable to because the program stops allowing the user to enter more numbers after 10. This is intentional and follows what the specification instructed. For this reason, it seems that this element of the program can be considered to work correctly.