//Student Name: Jing Ma

//Tuesday Labs 5:00

#include "library.h"

int mod(int const amount){

int result = 0;

if( amount %2 == 0){

return amount;

}

}

int factorial (int const amount) {

int result= 1;

if(amount>=1)

result= amount\* factorial(amount-1);

return result;

}

double power\_of\_y (double const amount, int const n){

int result = 1;

for (int i=1; i <= n; i++) {

result = amount \* result;

}

return result;

}

double plus\_minus (double const amount) {

if(amount==0)

return 0;

else {

double result = (-1)\*amount;

return result;

}

}

double another\_number(double const row, double const col){

// same as clicking function but there is a break funcion for = to go back to clicking function

double amount=0;

while(true) {

set\_pen\_color(color::dark\_grey);

fill\_rectangle(2\*row,2\*row, 56\*col, 8\*col);

set\_pen\_color(color::yellow);

move\_to(3\*row, 9\*col);

write\_string(amount);

wait\_for\_mouse\_click();

int const x = get\_click\_x();

int const y = get\_click\_y();

if (y>13\*row && y<22\*row) {

if (x>2\*col && x<11\*col){//7

amount= amount\*10+7;

print(amount);

}

else if (x>14\*col && x<23\*col){//8

amount= amount\*10+8;

print(amount);

}

else if (x>26\*col && x<35\*col){//9

amount= amount\*10+9;

print(amount);

}

else if (x>38\*col && x<47\*col){//plus

amount= amount + another\_number(row,col);

print(amount);

break;

}

else if (x>50\*col && x<59\*col){//power\_y

if (x<0)

amount;

else{

amount= power\_of\_y(amount, another\_number(row,col));

print(amount);

}

}

}

else if (y>25\*row && y<34\*row){

if (x>2\*col && x<11\*col){//4

amount= amount\*10+4;

print(amount);

}

else if (x>14\*col && x<23\*col){//5

amount= amount\*10+5;

print(amount);

}

else if (x>26\*col && x<35\*col){//6

amount= amount\*10+6;

print(amount);

}

else if (x>38\*col && x<47\*col){//minus

amount= amount - another\_number(row,col);

print(amount);

break;

}

else if (x>50\*col && x<59\*col){//+/-

amount= plus\_minus(amount);

print(amount);

}

}

else if (y>37\*row && y<46\*row){

if (x>2\*col && x<11\*col){//1

amount= amount\*10+1;

print(amount);

}

else if (x>14\*col && x<23\*col){//2

amount= amount\*10+2;

print(amount);

}

else if (x>26\*col && x<35\*col){//3

amount= amount\*10+3;

print(amount);

}

else if (x>38\*col && x<47\*col){//times

amount= amount \* another\_number(row,col);

}

else if (x>50\*col && x<59\*col){//mod

if(amount<0){

amount;

}

else{

amount= mod(amount);

print(amount);

}

}

}

else if (y>49\*row && y<58\*row){

if (x>2\*col && x<11\*col){//clear

amount= 0;

print(amount);

}

if (x>14\*col && x<23\*col){//0

amount= amount\*10+0;

print(amount);

}

if (x>26\*col && x<35\*col){//=

break;

}

if (x>38\*col && x<47\*col){//divide

if(x<0)

amount;

else{

amount= amount/another\_number(row,col);

print(amount);

}

}

if (x>50\*col && x<59\*col){//!

if (x<0)

amount;

else{

amount= factorial(amount);

print(amount);

}

}

}

new\_line();

}

return amount;

}

double clicking(double const row, double const col){

double amount=0;

while(true){

set\_pen\_color(color::dark\_grey);

fill\_rectangle(2\*row,2\*row, 56\*col, 8\*col);

set\_pen\_color(color::yellow);

move\_to(3\*row, 9\*col);

write\_string(amount);

wait\_for\_mouse\_click();

int const x = get\_click\_x();

int const y = get\_click\_y();

if (y>13\*row && y<22\*row){

if (x>2\*col && x<11\*col){//7

amount= amount\*10+7;

print(amount);

}

else if (x>14\*col && x<23\*col){//8

amount= amount\*10+8;

print(amount);

}

else if (x>26\*col && x<35\*col){ //9

amount= amount\*10+9;

print(amount);

}

else if (x>38\*col && x<47\*col){ //+

amount= amount + another\_number(row,col);

print(amount);

}

else if (x>50\*col && x<59\*col){ //power\_y

if (x<0)

amount;

else{

amount= power\_of\_y(amount,another\_number(row,col));

print(amount);

}

}

}

else if (y>25\*row && y<34\*row){//4

if (x>2\*col && x<11\*col){

amount= amount\*10+4;

print(amount);

}

else if (x>14\*col && x<23\*col){//5

amount= amount\*10+5;

print(amount);

}

else if (x>26\*col && x<35\*col){ //6

amount= amount\*10+6;

print(amount);

}

else if (x>38\*col && x<47\*col){ //-

amount= amount - another\_number(row,col);

print(amount);

}

else if (x>50\*col && x<59\*col){//+/-

amount= plus\_minus(amount);

print(amount);

}

}

else if (y>37\*row && y<46\*row){

if (x>2\*col && x<11\*col){//1

amount= amount\*10+1;

print(amount);

}

else if (x>14\*col && x<23\*col){ //2

amount= amount\*10+2;

print(amount);

}

else if (x>26\*col && x<35\*col){ //3

amount= amount\*10+3;

print(amount);

}

else if (x>38\*col && x<47\*col){ //\*

amount= amount \* another\_number(row,col);

}

else if (x>50\*col && x<59\*col){ //mod

if(amount<0)

amount;

else{

amount= mod(amount);

print(amount);

}

}

}

else if (y>49\*row && y<58\*row){

if (x>2\*col && x<11\*col){ //c

amount= 0;

print(amount);

}

if (x>14\*col && x<23\*col){ //0

amount= amount\*10+0;

print(amount);

}

if (x>26\*col && x<35\*col){ //=

amount=amount;

print(amount);

}

if (x>38\*col && x<47\*col){//divide

if(x<0)

amount;

else{

amount= amount/another\_number(row,col);

print(amount);

}

}

if (x>50\*col && x<59\*col){//!

if (x<0)

amount;

else{

amount= factorial(amount);

print(amount);

}

}

}

new\_line();

}

}

void draw\_char(double const row[], double const column[], double const size, int const i, int const j){

set\_pen\_color(color::black);

set\_font\_size(size);

double const char\_pos\_x = row[i]+.25\*size;

double const char\_pos\_y = column[j]+.8\*size;

move\_to(char\_pos\_x, char\_pos\_y);

if (j==1 && i==0) write\_char('7');

else if (j==1 && i==0) write\_char('7');

else if (j==1 && i==1) write\_char('8');

else if (j==1 && i==2) write\_char('9');

else if (j==1 && i==3) write\_char('+');

else if (j==1 && i==4){

set\_font\_size(30);

write\_char(L'X');

write\_char(L'^');

write\_char(L'y');

}

else if (j==2 && i==0) write\_char('4');

else if (j==2 && i==1) write\_char('5');

else if (j==2 && i==2) write\_char('6');

else if (j==2 && i==3) write\_char('-');

else if (j==2 && i==4) {

set\_font\_size(50);

write\_char(L'+');

write\_char(L'/');

write\_char(L'-');

}

else if (j==3 && i==0) write\_char('1');

else if (j==3 && i==1) write\_char('2');

else if (j==3 && i==2) write\_char('3');

else if (j==3 && i==3) write\_char(L'×');

else if (j==3 && i==4){

set\_font\_size(30);

write\_char(L'm');

write\_char(L'o');

write\_char(L'd');

}

else if (j==4 && i==0) write\_char('c');

else if (j==4 && i==1) write\_char('0');

else if (j==4 && i==2) write\_char('=');

else if (j==4 && i==3) write\_char(L'÷');

else if (j==4 && i==4) write\_char('!');

}

void draw\_buttons(double const row[],double const column[], double const size){

double const button\_size= 9\*size;

double const display\_size= 57.0\*size;

for(int i=0; i<5; i++){

for(int j=0; j<5; j++){

if(j==0){

if(i==0){

set\_pen\_color(color::dark\_grey);

fill\_rectangle(column[j]-1, row[i]-1, display\_size+2, button\_size+2);

set\_pen\_color(color::dark\_grey);

fill\_rectangle(column[j], row[i], display\_size, button\_size);

}

}

else{

set\_pen\_color(color::black);

fill\_rectangle(column[i]-1, row[j]-1, button\_size+2, button\_size+2);

set\_pen\_color(color::light\_grey);

fill\_rectangle(column[i], row[j], button\_size, button\_size);

draw\_char(column, row, button\_size,i,j);

}

}

}

}

void button\_array(double const size){

double const button\_size= size/61;

double const row\_pos= size/60.0; //60 is the size factor of the calc's height

double const col\_pos = size/61.0;//61 is the size factor of the calc's width

double const row\_start\_pos[5]={2\*row\_pos, 13\*row\_pos, 25\*row\_pos, 37\*row\_pos, 49\*row\_pos};

double const column\_start\_pos[5]={2\*col\_pos, 14\*col\_pos, 26\*col\_pos, 38\*col\_pos, 50\*col\_pos};

draw\_buttons(row\_start\_pos, column\_start\_pos, button\_size);

clicking(row\_pos, col\_pos);

}

void main(){

int const size=500;

double const x=size;

double const y=.9836\*size; // 60:61 is the calculator proportions of y:x

make\_window(x,y);

set\_pen\_color(color::white);

fill\_rectangle(0.0,0.0,x,y);

button\_array(size);

}

A close up of a keyboard

Description automatically generatedA close up of a keyboard

Description automatically generated