//eliska vrzalova

//date: 10/10/2023

//est time 10 min

//actual time 12 min

//The Fly-Sugar problem

//calculation of fly’s route using Phytagorean theorem

//variables, all float in C++

lenght = 0; //room measurement

width = 0; //room measurement

height = 0; //room measurement

heightForFly = 0; //height (diagonal) needed to calculate fly’s route

route = 0; //the distance the fly will fly

//ask user for input

Display message “The length of the room in meters is:”;

input length;

Display message “The width of the room in meters is:”;

input width;

Display message “The height of the room in meters is:”;

input height;

//calculations

heightForFly = square root (height\*height + width\*width); //calculates the diagonal needed for the calculation of the fly’s route

route = square root (height\*height + heightForFly\*heightForFly); //calculates the distance the fly will fly

//display answer

Display message “The shortest route the fly will fly is” + route + “meters.”;

//eliska vrzalova

//date: 10/10/2023

//est time 10 min

//actual time 13 min

//Medicine dispensing problem

//calculates the number of injections taken from the bottle and left over mililitres

//variables

litres = 0; //float in C++

mililitres = 0; //float in C++

L\_TO\_ML = 1000; //constant in C++

injections = 0; //how many injections are there in the bottle

leftOverMl = 0; //left over mililitres after all injections taken

//ask user for input

Display message “How many liters are there in a bottle of medicine?”;

input litres;

Display message “How many mililitres are in 1 injection?”;

input injections;

//calculations

litres = litres8\*L\_TO\_ML; //converts litres to mililitres

injections = litres/mililitres; //calculates the injections in the bottle

leftOverMl = litres – (injections\*mililitres); //calculates the mililitres left

//display answer

Display message “The number of injections in this bottle is” + injections;

Display message “There is/are” + leftOverMl + “mililitre(s) left over after all injections have been taken.”;

//eliska vrzalova

//date: 10/10/2023

//est time 10 min

//actual time 15 min

//Income tax

//calculates the net pay and tax which is due after the gross income is input by the user

//variables

grossIncome = 0;

netPay = 0;

taxDue = 0;

TAX\_BAND = 22; //constant in C++, tax band is se to be 22%

TAX\_FREE\_ALLOWANCE = 12000; //everything under 12000 euro is not taxed

//ask user for input

Display message “What is the gross income?”;

input grossIncome;

//calculations

IF (grossIncome > 12000)

taxDue = ((grossIncome – TAX\_FREE\_ALLOWANCE)\*TAX\_BAND)/100; //calculates the tax which is due if the gross income is over 12000 euro

netPay = grossIncome – taxDue; //calculates the net pay if the gross income is over 12000 euro

ELSE

taxDue = 0; //there is no tax if the income is under 12000 euro

netPay = grossIncome ;//calculates the the net pay when the income is under 12000 euro

ENDIF;

//answer

Display message “The net pay is” + netPay + “ euro and the tax which is due is” + taxDue + “euro.”;

//eliska vrzalova

//date: 10/10/2023

//est time 10 min

//actual time 10 min

//Leg counting problem

//calculates the number of sheep and chickens after the number of leg sis input by user

//variables, all int in C++

animals = 0;

legs = 0;

sheep = 0;

chickens = 0;

//ask user for input

Display message “What is the total number of animals?”;

input animals;

Display message “What is the total number of legs?”;

input legs;

//calculations

sheep = (legs – 2\*animals)/2; //calculates the number of sheep

chickens = animals – sheep //calculates the number of chickens

//answer

Display message “There are “ + sheep + “sheep and” + chickens + “chicken(s) in the farmyard;

//eliska vrzalova

//date: 10/10/2023

//est time 20 min

//actual time 25 min

//Ordering system

//calculates the cost of the order in Euro, Dollars and calculates the cost of ordered but not used materiál in Dollars

//variables

girders = 0;

GIRDER\_PRICE = 6; //constant in C++, price per 1 girder

totalDollars = 0;

totalEuro = 0;

EURO\_TO\_DOLLAR = 0,9; //float const in C++, 1 dollar is 90 cents

notUsed = 0;

numberGroup = 0;

numberGroupRounded = 0;

groups = 0; // needed for the calculation for the number of groups in else statement

GROUP = 10; //constant in C++, number of girders in 1 group

GROUP\_PRICE = 60; //Price of 1 group in dollars

//ask user for input

Display message “How many girders do you need?”;

input girders;

//calculations

numberGroup = girders/GROUP; //calculates the number of groups costumer needs to buy

numberGroupRounded = round(girders/GROUP); ////calculates the number of groups costumer needs to buy but rounded, using floor in C++ to round it down

IF(number group == numberGroupRounded);

totalDollars = GROUP\_PRICE \* numberGroup; //calculates the cost of the order in dollars

totalEuro = totalDollars\*EURO\_TO\_DOLAR; //calculates the cost of the order in Euro

notUsed = totalDollars – (girders – GIRDER\_PRICE); //calculates the cost of ordered but not used girders

ELSE

groups = numberGroup + 1; //calculates the number of groups

totalDollars = GROUP\_PRICE \* groups; //calculates the cost of the order in dollars

totalEuro = totalDollars\*EURO\_TO\_DOLAR; //calculates the cost of the order in Euro

notUsed = totalDollars – (girders – GIRDER\_PRICE); //calculates the cost of ordered but not used girders

ENDIF;

//display answer

Display message “Total cost in Dollars is:” + totalDollars;

Display message “Total cost in Euro is:” + Euro;

Display message “Total cost in Dollars of girders you have ordered but you do not need them is:” + notUsed;

//eliska vrzalova

//date: 10/10/2023

//est time 20 min

//actual time

//Exam grades

//variables

examMark= 0;

FAIL = 40;

PASS = 55;

MERIT = 70;

DISTINCTION = 100;

//input

display message"What is your exam mark?";

input examMark;

//checking the grade

if(examMark >= 0 and examMark < FAIL)

display message"Student has failed."; //range of fail 0 - 39

else if(examMark >= FAIL and examMark <PASS) //range of pass 40 - 54

display message"Student has passed.";

else if(examMark >= PASS and examMark < MERIT) //range of merit 55-69

display message"Student has a merit.";

else if(examMark >= MERIT and examMark <=DISTINCTION) //range of distinction 70-100

display message"Student has distinction.";

else

display message"Error- exam mark has an incorrect value."; //number entered is out of 0-100 range

end else if

//eliska vrzalova

//date: 10/10/2023

//est time 10 min

//actual time 12 min

//The Ant-Sugar problem

//calculation of ant’s route using Phytagorean theorem

//variables, all float in C++

lenght = 0; //room measurement

width = 0; //room measurement

height = 0; //room measurement

diagonal = 0; //part of ants rout

route = 0; //the distance the ant will crawl

//ask user for input

Display message “The length of the room in meters is:”;

input length;

Display message “The width of the room in meters is:”;

input width;

Display message “The height of the room in meters is:”;

input height;

//calculations

diagonal = square root (lenght\*lenght + width\*width); //calculates the diagonal which is part of the ant’s route

route = height + diagonal; //calculates the route the ant will crawl

//display answer

Display message “The shortest route the ant will crawl is” + route + “meters.”;