Semester 2 Prac1 –

Part1

Q1

### Which of the following represent valid identifiers?

a) Abc e) €€€ i) a\_1

b) MH f) 25or6to4 j) customerNumber

c) while g) first time

d) strings h) second-Name

**Q2**

int sum = 0;

int x = 0;

while (x < 25)

{

sum += x + 5;

x+=3;

}

|  |
| --- |
| What value will sum have after the above is executed? |

Rewrite by hand the above using a for statement

Q3

Below is a piece of code to sum all the integer values entered by a user, the series is terminated with the sentinel value -999. It contains a logical error and a number of potential run-time errors. Describe the outcome of these errors and suggest how they may be fixed.

int examScore;

int total= 0,count = 0;

Console.Write("Enter score : ");

examScore = int.Parse(Console.ReadLine());

while (examScore != -999)

{

Console.Write("Enter score : ");

examScore = int.Parse(Console.ReadLine());

total += examScore;

count++;

}

double avg = total / count;

Console.WriteLine("average is {0}", avg);

Part 2

We have come across some of these before, this is a review

1. Write a *for* loop to print all integers between 1 and 20 in ascending order. Now write it with a *while* loop.
2. Write a *for* loop to print all integers between 1 and 20 excluding 6 and 8 in ascending order – implement with the continue keyword.
3. Write a *for* loop to print all integers between 1 and 20 in descending order.
4. Write a *for* loop to sum all the odd numbers between the integers n1 and n2, where n1 and n2 are entered by the user. Assume n1 <=n2. If the sum exceeds 500 print a message “sum too large” and exit the loop.
5. Write a program to determine much will be in your regular savings account after a number of years at a fixed monthly interest rate of 1%. Inputs will be initial balance, number of years saving and monthly saving amount. (test with initial balance = 1000, years = 2; monthly saving = 100)

What kind of loop would work best here?

Modify the program, so that that the balance after each month is printed.

Q6

An internet cafe charges a €2.00 minimum fee to use the internet for up to three hours. The cafe charges an additional €0.50 per hour for each additional hour or part thereof. The maximum charge for any given session is €10.00. .

Write a program that calculates and displays the charges for each customer who uses the cafe. Your program will accept the hours used by each customer. The program will display the charge for each customer and a running total for all usage. The program will have at least one method that calculates the charge for each customer.

**Sample Dialoge:**

*Enter number of hours (-999 to quit) : 2*

*Customer Charge : 2.00 Total Receipt 2.00*

*Enter number of hours (-999 to quit) : 3*

*Customer Charge : 2.00 Total Receipt 4.00*

*Enter number of hours (-999 to quit) : 4*

*Customer Charge : 2.50 Total Receipt 6.50*

*Enter number of hours (-999 to quit) : 5*

*Customer Charge : 3.00 Total Receipt 9.50*

*Enter number of hours (-999 to quit) : 10*

*Customer Charge : 5.50 Total Receipt 15.00*

*Enter number of hours (-999 to quit) : 20*

*Customer Charge : 10.00 Total Receipt 25.00*

*Enter number of hours (-999 to quit) : 21*

*Customer Charge : 10.00 Total Receipt 35.00*

*Enter number of hours (-999 to quit) : -999*

**Part 3**

**Nested Control**

**Q1**

A year with 366 days is called a leap year. A is a leap year if it is divisible by four, except that it is not a leap year if it is divisible by 100; however it is a leap year if it is divisible by 400. There were no exceptions before the introduction of the Gregorian year in 1582. Write a program that asks the user for a year and computes whether that year is a leap year.

Q2

Write a program to print the pattern. Printing one ‘#’ at a time

#

# #

# # #

# # # #

# # # # #

Do it now upside down

Q3

An on line book retailer charges a flat 2.50 for each delivery, on top of which it applies the following the progressive fee rate depending on delivery weight. The progressive fee rate is applicable to the book total weight.

|  |  |  |
| --- | --- | --- |
| **Type of delivery** | **For weight of Books** | **Fee Rate (Euro/gram)** |
| Regular | 0 - 2000 grams | 0.025 |
| Regular | 2001 – 5000 grams | 0. 03 |
| Regular | 5001 and over | 0.05 |
| Express | Use the same rate as Regular delivery plus 1.50 euro extra fee | |
| Super Express | Use the same rate as Regular delivery plus 2.50 | |
| Super Super Express | Use the same rate as Regular delivery plus 3.50 | |

**Example:**

If the total weight of books is 3500 grams, we can calculate the fee as follows.

Fee = 2.5 + ( 2000 grams \* 0.0250) + ( 3500 – 2000 grams) \* 0.03

= 97.50 Euro  
Write a program to compute the total fee for delivery of books, the user shall be able to calculate the fee as many times as they wish.. The input will be the total weight of books in grams and the Delivery type (R for Regular Delivery and X for Express Delivery, SX, SSX)

Example Program Dialogue:

Enter Weight of book (grams) :3500

Enter Delivery (x/r/sx/ssx) : r

You have to pay 97.5 0 euro for 3500 grams

Calculate another (y / n) :

Q4

Write a program that computes taxes for the following schedule:

|  |  |  |  |
| --- | --- | --- | --- |
| If your status is single and if the taxable income is over | But not over | The tax is | Of the amount over |
| €0 | €8,000 | 10% | €0 |
| €8,000 | €32,000 | €800+15% | €8,000 |
| €32,000 |  | €4,400 + 25% | €32,000 |
| If your status is married and if the taxable income is over | But not over | The tax is | Of the amount over |
| €0 | €16,000 | 10% | €0 |
| €16,000 | €64,000 | €1,600+15% | €16,000 |
| €64,000 |  | €8,800 + 25% | €64,000 |

Part 4

Input validation

Sometime users don’t always do what we ask them to do. For example we might as a user to enter a number in the range 0 to 19.

They might enter a number outside this range

They might enter nothing

They might enter a non numeric character

There are ways of developing program that tolerate bad input (exception handlng). For now our goal is simply to detect bad input and to exit the program when it occurs.

int input;

bool validInput = false;

validInput = int.TryParse(Console.ReadLine(), out input);

if (!validInput )

{

Console.WriteLine("Please enter a number between 0 and 20");

return;

}

If (input < 0 && input > 20)

{

Console.WriteLine("Please enter a number between 0 and 20");

return

}

How might you modify this to allow repeated attempts?