

## Pizzeria Pepe:

Un programa controla si una comanda de pizzas la pot portar un repartidor.

Heu de tenir en compte que un repartidor pot portar 10 pizzas com a màxim a la moto.

El programa retorna true si la comanda es pot portar i false en cas contrari.

Es demana:

- Fer el codi font del programa.

```
public class pizzeriaPepe {  
  
    public static boolean potRepartir(int pizzas){  
  
        boolean resultat = false;  
  
        if(pizzas>=1 && pizzas<=10){  
  
            resultat = true;  
  
        }  
  
        return resultat;  
  
    }  
  
}
```

- Fer la taula amb les particions equivalents i casos vàlids i no vàlids.

| Paràmetre entrada | Regla a aplicar                       | Classes vàlides                   | Classes no vàlides  |
|-------------------|---------------------------------------|-----------------------------------|---|
| pizzas            | És un número? + rang<br>valors (1-10) | 1. $1 \leq \text{pizzas} \leq 10$ | 2. $\text{pizzas} < 1$<br>3. $\text{pizzas} > 10$<br>4. pizzas no és un número. |

| Pizzas | Classe coberta | Resultat |
|--------|----------------|----------|
| 1      | 1              | true     |
| -1     | 2              | false    |
| 12     | 3              | false    |
| cinc   | 4              | ERROR    |

- Fer la taula amb l'anàlisi de valors límit i casos vàlids i no vàlids.

| Paràmetre d'entrada | Regla a aplicar                    | Classes vàlides   | Classes no vàlides           |
|---------------------|------------------------------------|---|------------------------------|
| pizzas              | És un número? + rang valors (1-10) | 5. pizzas=1<br>6. pizzas=2<br>7. pizzas=9<br>8. pizzas=10 | 9. pizzas=0<br>10. pizzas=11 |

| Pizzas | Classe coberta | Resultat |
|--------|----------------|----------|
| 1      | 5              | true     |
| 2      | 6              | true     |
| 9      | 7              | true     |

|    |    |       |
|----|----|-------|
| 10 | 8  | true  |
| 0  | 9  | false |
| 11 | 10 | false |

## Transports Jean Claude:

Un programa gestiona si una càrrega es pot portar amb una furgoneta.

Una càrrega sempre pesarà almenys 500 kg. per aprofitar el transport, però no podrà pesar més de 900kg.

Una furgoneta ha de tenir la capacitat de portar un pes mínima de 500kg i màxima de 750kg.

Si una càrrega no es pot portar el programa retorna -1, en cas contrari retorna 0.

Es demana:

- Fer el codi font del programa.

```
public class TransportsJeanClaude {
```

```
    public static int potPortar(int carrega, int capacitat){
```

```
        int resultat = -1;
```

```
        if(carrega>=500 && carrega<=900){
```

```
            if(capacitat>=500 && capacitat<=750){
```

```
                if(capacitat>=carrega){
```

```
                    resultat = 0;
```

```
                }
```

```
            }
```

```
        }
```

```

    return resultat;
}
}

```

- Fer la taula amb les particions equivalents i casos vàlids i no vàlids.

| Paràmetre entrada | Regla a aplicar                       | Classes vàlides                         | Classes no vàlides  |
|-------------------|---------------------------------------|---|---|
| carrega           | És un número? + rang valors (500-900) | 1. $500 \leq \text{carrega} \leq 900$   | 2. $\text{carrega} < 500$<br><br>3. $\text{carrega} > 900$<br><br>4. carrega no és un número.<br><br>5. $\text{carrega} > \text{capacitat}$ |
| capacitat         | És un número? + rang valors (500-750) | 6. $500 \leq \text{capacitat} \leq 750$ | 7. $\text{capacitat} < 500$<br><br>8. $\text{capacitat} > 750$<br><br>9. capacitat no és un número.   |

| carrega | capacitat | Classe coberta | Resultat |
|---------|-----------|----------------|----------|
| 600     | 700       | 1,6            | 0        |
| 300     | 700       | 2,6            | -1       |

|            |     |       |               |
|------------|-----|-------|---------------|
| 1000       | 700 | 3,6   | -1            |
| cinc-cents | 700 | 4,6   | -1 o<br>ERROR |
| 800        | 700 | 5,6,1 | -1            |
| 500        | 400 | 7,1   | -1            |
| 500        | 800 | 8,1   | -1            |
| 500        | mil | 9,1   | -1 o<br>ERROR |

- Fer la taula amb l'anàlisi de valors límit i casos vàlids i no vàlids.

| <b>Paràmetre<br/>d'entrada</b> | <b>Regla a aplicar</b>   | <b>Classes vàlides</b>   | <b>Classes no<br/>vàlides</b>          |
|--------------------------------|--|--|--|
| carrega                        | És un número? +<br>rang valors<br>(500-900) + carrega<br>< capacitat | 10. carrega=500<br>11. carrega=501<br>12. carrega=899<br>13. carrega=900         | 14. carrega=499<br>15. carrega=901     |
| capacitat                      | És un número? +<br>rang valors<br>(500-750) + carrega<br>< capacitat | 16. capacitat=500<br>17. capacitat=501<br>18. capacitat=749<br>19. capacitat=750 | 20. capacitat=499<br>21. capacitat=751 |

| <b>carrega</b>   | <b>capacitat</b> | <b>Classe coberta</b> | <b>Resultat</b> |
|------------------|------------------|-----------------------|-----------------|
| 500              | 500              | 10                    | 0               |
| 501              | 500              | 11                    | 0               |
| 899              | 500              | 12                    | 0               |
| 900              | 500              | 13                    | 0               |
| 499              | 500              | 14                    | -1              |
| 901              | 500              | 15                    | -1              |
| <b>capacitat</b> | <b>carrega</b>   | <b>Classe coberta</b> | <b>Resultat</b> |
| 500              | 500              | 16                    | 0               |
| 501              | 500              | 17                    | 0               |
| 749              | 500              | 18                    | 0               |
| 750              | 500              | 19                    | 0               |
| 499              | 500              | 20                    | -1              |
| 751              | 500              | 21                    | -1              |

## Control de temperatura:

Un programa gestiona el modificador de temperatura del sistema de calefacció d'un restaurant.

Aquest programa rep la medició de la temperatura del restaurant en graus celsius, acceptant com entrades vàlides de -10 a 50 graus. (medidor)

A més té una entrada d'usuari/a amb la temperatura que es vol mantenir, que va de 15 a 40 graus. (termostat).

El sistema té tres sortides possibles que representen la potencia del sistema de calefacció: 0, 1, 2.

Segons la informació que té en cada moment el programa farà el següent:

Si la temperatura del medidor és més alta que la del termostat, la potencia del sistema serà 0. Si la temperatura del medidor és més baixa o més alta que la del termostat però només entre 0 i 2 graus, la potència serà 1. Si la temperatura del medidor és més baixa que la del termostat en més de dos graus, la potència del sistema serà 2.

Es demana:

- Fer el codi font del programa.

```
public static int modificadorTemperatura(int medidor, int termostat){
```

```
    //variable que retorna la funció:
```

```
    int resultat = -1;
```

```
    //Si la temperatura del medidor és més alta que la del termostat, la potencia del sistema serà 0.
```

```
    if(medidor>termostat){ resultat=0;}
```

```
    //Si la temperatura del medidor és més baixa o més alta que la del termostat però només entre 0 i 2 graus, la potència serà 1.
```

```
    int variableAuxiliar = Math.abs(termostat-medidor);
```

```
    if(variableAuxiliar>=0 && variableAuxiliar<=2){
```

```
        resultat = 1;
```

```
    }
```

//Si la temperatura del medidor és més baixa que la del termostat en més de dos graus, la potència del sistema serà 2.

```
variableAuxiliar = medidor - termostat;
```

```
if(variableAuxiliar<-2){
```

```
    resultat = 2;
```

```
}
```

// Aquest programa rep la medició de la temperatura del restaurant en graus celsius, acceptant com entrades

// vàlides de -10 a 50 graus. (medidor).

// A més té una entrada d'usuari/a amb la temperatura que es vol mantenir, que va de 15 a 40 graus. (termostat).

```
if(medidor<-10 || medidor>50 || termostat<15 || termostat>40) {
```

```
    resultat = -1; //-1 indica que no es compleixen les condicions d'entrada.
```

```
}
```

```
return resultat;
```

```
}
```

- Fer la taula amb les particions equivalents i casos vàlids i no vàlids.

| Paràmetre entrada | Regla a aplicar                        | Classes vàlides         | Classes no vàlides                      |
|-------------------|--|-------------------------|---|
| medidor           | És un número? + rang valors (-10 a 50) | 1. -10 <= medidor <= 50 | 2. medidor < -10<br><br>3. medidor > 50 |



| Paràmetre entrada | Regla a aplicar                       | Classes vàlides                       | Classes no vàlides  |
|-------------------|---------------------------------------|---------------------------------------|---|
|                   |                                       |                                       | 4. medidor no és un número.   |
| termostat         | És un número? + rang valors (15 a 40) | 5. $15 \leq \text{termostat} \leq 40$ | 6. termostat < 15<br><br>7. termostat > 40<br><br>8. termostat no és un número. |

| medidor | termostat | Classe coberta | Resultat   |
|---------|-----------|----------------|------------|
| 20      | 20        | 1,5            | 0          |
| -16     | 20        | 2,5            | -1         |
| 55      | 20        | 3,5            | -1         |
| vint    | 20        | 4,5            | -1 o error |
| 20      | 10        | 6,1            | -1         |
| 20      | 50        | 7,1            | -1         |
| 20      | vint      | 8,1            | -1 o error |

- Fer la taula amb l'anàlisi de valors límit i casos vàlids i no vàlids.

| Paràmetre entrada | Regla a aplicar                        | Classes vàlides  | Classes no vàlides                   |
|-------------------|--|--|--------------------------------------|
| medidor           | És un número? + rang valors (-10 a 50) | 9. medidor=-10<br>10. medidor=-9<br>11. medidor=49<br>12. medidor=50         | 13. medidor=-11<br>14. medidor=51    |
| termostat         | És un número? + rang valors (15 a 40)  | 15. termostat=15<br>16. termostat=16<br>17. termostat=39<br>18. termostat=40 | 19. termostat=14<br>20. termostat=41 |

| medidor   | termostat | Classe coberta | Resultat |
|-----------|-----------|----------------|----------|
| -10       | 20        | 9              | 2        |
| -9        | 20        | 10             | 2        |
| 49        | 20        | 11             | 0        |
| 50        | 20        | 12             | 0        |
| -11       | 20        | 13             | -1       |
| 51        | 20        | 14             | -1       |
| termostat | medidor   | Classe coberta | Resultat |
| 15        | 20        | 15             | 0        |

|    |    |    |    |
|----|----|----|----|
| 16 | 20 | 16 | 0  |
| 39 | 20 | 17 | 2  |
| 40 | 20 | 18 | 2  |
| 14 | 20 | 19 | -1 |
| 41 | 20 | 20 | -1 |

## Activitats debug:

### Factorial:

El factorial d'un nombre és el resultat de multiplicar el número per ell mateix -1 tantes vegades com siguin necessàries fins arribar a 1.

Per exemple el factorial de 5 és:

$$5 * 4 * 3 * 2 * 1$$

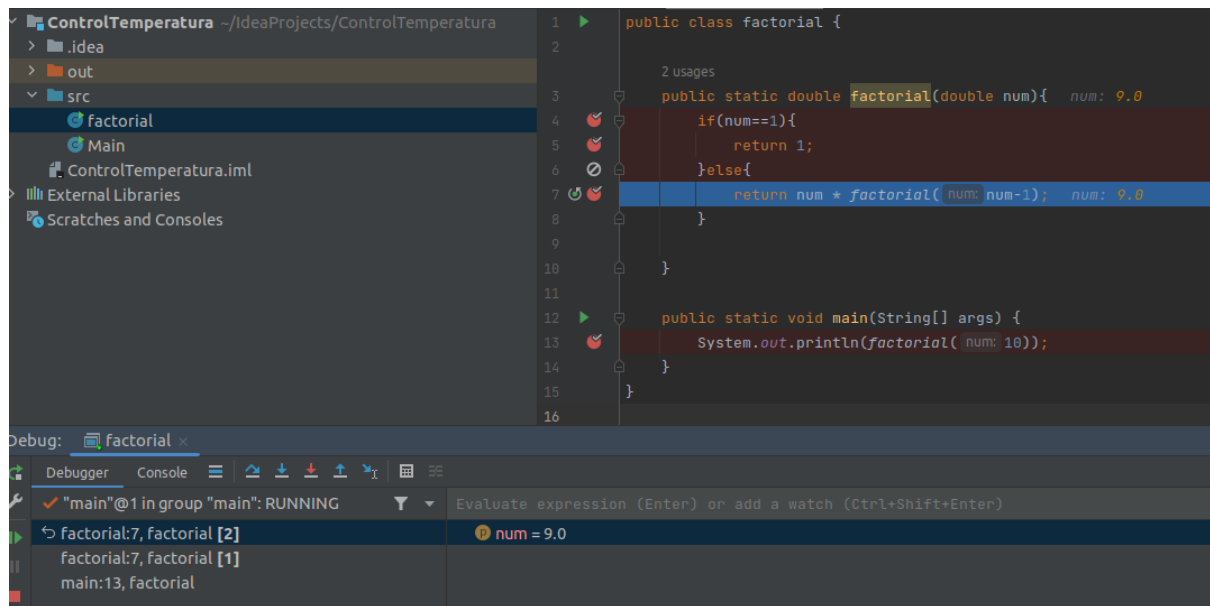
Fes una funció factorial que rebi un número com paràmetre i retorni el seu factorial.

Es demana:

- Codi del programa.

```
public static double factorial(double num){
    if(num==1){
        return 1;
    }else{
        return num * factorial(num-1);
    }
}
```

- Captura de pantalla amb un punt d'interrupció que deixi veure totes les crides a la funció (agafeu un valor menor a 10).



## Taula de multiplicar:

Fes un programa que crea una matriu de números del 1 al 10. Aquest programa rep per argument d'entrada un número sencer i retorna per terminal la taula de multiplicar d'aquest número multiplicant el argument per cada valor de la matriu.

Es demana:

- Codi del programa.

```
public class Taulamultiplicar {
```

```
    public static void main(String[] args) {
```

```
        System.out.println("La taula de multiplicar del "+args+" es:");
```

```
        mostraMultiplicacio(args);
```

```
    }
```

```

private static void mostraMultiplicacio(String[] input){

    //matriu amb els números de l'1 al 10

    int[] taula = new int[10];

    for (int i = 0; i < taula.length; i++) {

        taula[i] = i+1;

    }

    int result = 0;

    for (String s :input){

        for (int i = 0; i < taula.length; i++) {

            result = taula[i]* Integer.parseInt(s);

            System.out.println(result);

        }

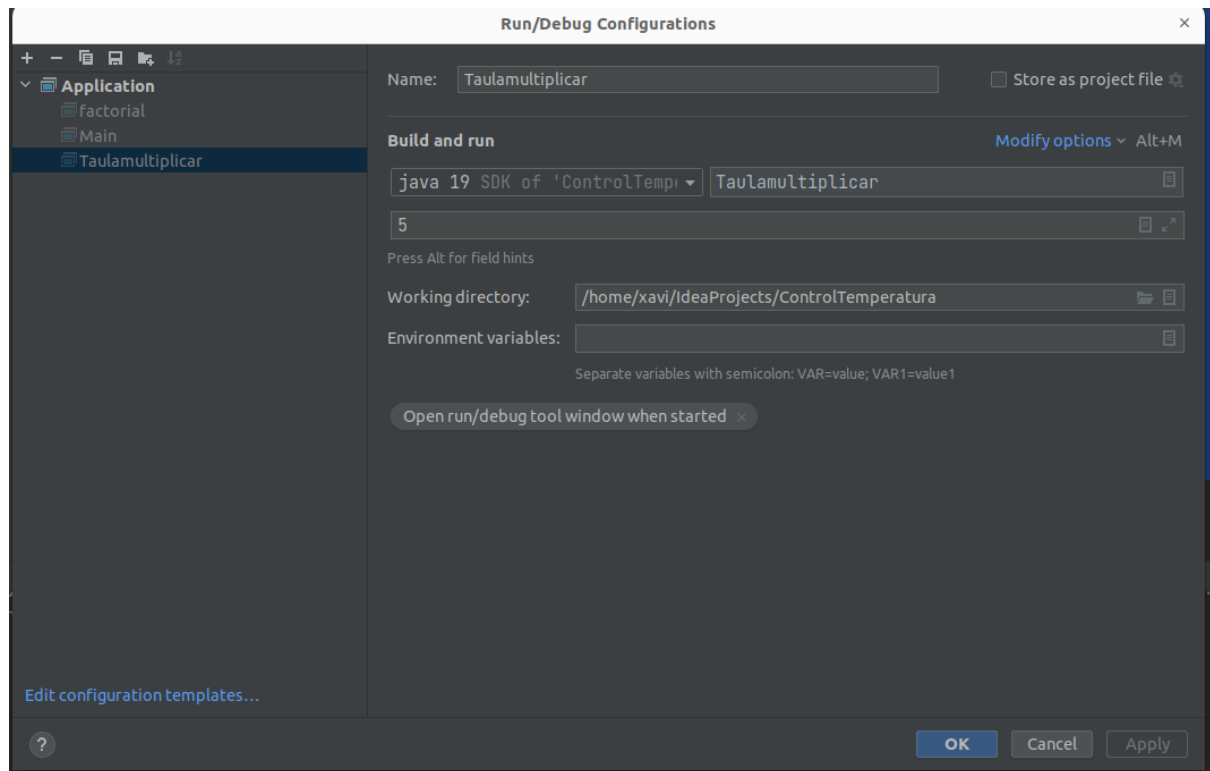
    }

}

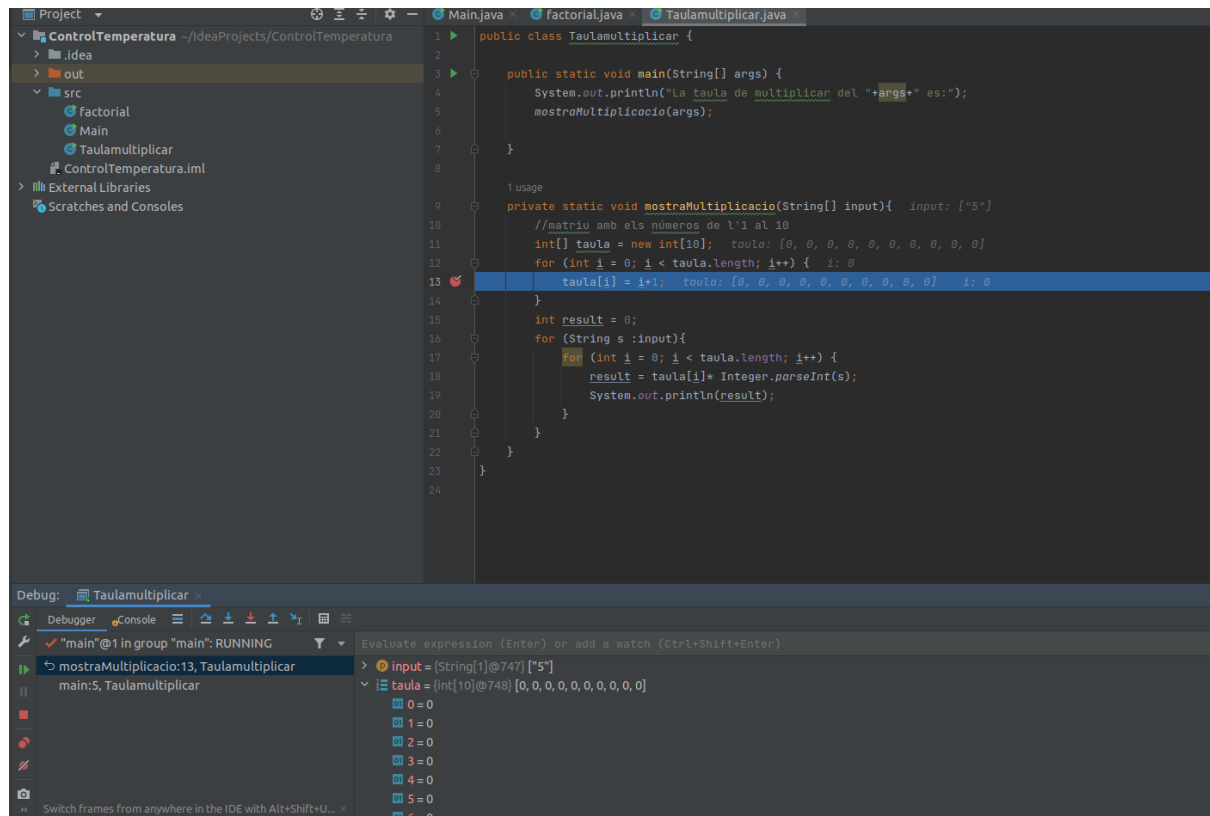
```

- Captura de pantalla de com li passeu a IntelliJ com argument del programa un número. (Mireu exemple findAverage).

Posem un 5 com paràmetre.



- Captura de com feu un punt d'interrupció al bucle de creació de la matriu i mostreu els valors de la matriu.



- Captura de punt d'interrupció al bucle de multiplicació i com modifiquem a mà els valors de la matriu de números per a que l'execució retorni el número 1 10 vegades quan l'argument d'entrada era 1.

