Hedi Ben Abid
300123192

Pierre Akladios
300114467

Tout le code est disponible sur le repo suivant :
https://github.com/Hadhoud5/seg3503_playground/tree/main/Assignment_2

Partie 1 :

https://github.com/Hadhoud5/seg3503_playground/tree/main/Assignment_2/Partie1

Q 1.1
Voir sur github (Partie 1)

Q 1.2
Expected Results: Percentage_grade/Letter_grade/Numeric_grade

Test Case Number	Test Data	Expected Results	Conditions Covered	Branches Covered

1	[],[],0.4,0.3	017/"EIN"/0	if Enum.count(homework) == 0 do
			
			if Enum.count(labs) == 0 do
			
			if Enum.count(homework) == 0 do
			
			if Enum.count(labs) == 0 do
			
			|> Enum.reject(fn mark -> mark < 0.25 end)
				True Branch for all the conditions mentioned in the previous columns

			if avg_homework < 0.4 || avg_exams < 0.4 || num_labs < 3 do	
			if Enum.count(homework) == 0 do	
			if Enum.count(labs) == 0 do	
			|> Enum.reject(fn mark -> mark < 0.25 end)	
			if avg_homework < 0.4 || avg_exams <	

			0.4 || num_labs < 3 do	
2	[0.9,1],[0.75,1,1, 0.25],0.6,0.5	71/"B"/6	if Enum.count(homework) == 0 do	F
			Else	T
			if Enum.count(labs) == 0 do	F
			Else	T
			if Enum.count(homework) == 0 do	F
			Else	
			if Enum.count(labs) == 0 do	T
			Else	F
			if avg_homework < 0.4 || avg_exams < 0.4 || num_labs < 3 do	T
				F
			Else	
			cond do #22 mark > 0.895 -> "A+" #23 for all the options	T
			mark > 0.845 -> "A"	F
			mark > 0.795 -> "A-"	F
			mark > 0.745 -> "B+"	F
			mark > 0.695 -> "B"	F

			if Enum.count(homework) == 0 do	T
				F
			Else	
			if Enum.count(labs) == 0 do	T
				F
			Else	
			if avg_homework < 0.4 || avg_exams < 0.4 || num_labs < 3 do	T
				F
			Else	

			cond do #22
			    mark > 0.895 -> 10 #23 for all options
			    mark > 0.845 -> 9
			    mark > 0.795 -> 8
			    mark > 0.745 -> 7
			    mark > 0.695 -> 6
				T

				F
				F
				F
				F
				T

3	[0.3,1,0],[0.75,1,1,0.25,0,1], 0.6, 0.45	52/"D"/2	if Enum.count(homework) == 0 do
			Else
			if Enum.count(labs) == 0 do
				F

				T
			Else	F

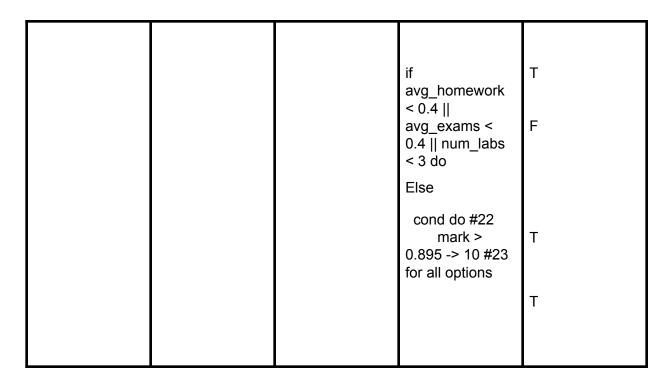|				

			if Enum.count(homework) == 0 do
			
			Else
				T
				
				F

			if Enum.count(labs) == 0 do
			
			Else
				T
				
				F

			if avg_homework < 0.4 || avg_exams < 0.4 || num_labs < 3 do
			
			 Else
				T
				
				F

			cond do #22
			    mark > 0.895 -> "A+" #23 for all the options
			    mark > 0.845 -> "A"
			    mark > 0.795 -> "A-"
			    mark > 0.745 -> "B+"
			    mark > 0.695 -> "B"
			    mark > 0.645 -> "C+"
			    mark > 0.595 -> "C"
			    mark > 0.545 -> "D+"
			    mark > 0.495 -> "D"
				T
				
				F
				
				F
				F
				F
				F
				F
				F
				F

			if Enum.count(homework) == 0 do
			
			Else
				T
				
				F
				
				T

			if Enum.count(labs) == 0 do	T
			Else	F
			if avg_homework < 0.4 || avg_exams < 0.4 || num_labs < 3 do	T
				F
			Else	

			cond do #22
			    mark > 0.895 -> 10 #23 for all options
			    mark > 0.845 -> 9
			    mark > 0.795 -> 8
			    mark > 0.745 -> 7
			    mark > 0.695 -> 6
			    mark > 0.645 -> 5
			    mark > 0.595 -> 4
			    mark > 0.545 -> 3
			    mark > 0.495 -> 2
				T

				F
				F
				F
				F
				F
				F
				F
				F
				T

4	[1,1,1], [1,1,1,1,1], 1, 1	100/"A+"/10	if Enum.count(homework) == 0 do	F
			Else	T
			if Enum.count(labs) == 0 do	F
			Else	T
			if Enum.count(homework) == 0 do	F
			Else	
			if Enum.count(labs) == 0 do	T
				F
			Else	
			if avg_homework < 0.4 || avg_exams < 0.4 || num_labs < 3 do	T
				F
			Else	
			cond do #22 mark > 0.895 -> "A+" #23 for all the options	T
				T
			if Enum.count(homework) == 0 do	T
				F
			Else	
			if Enum.count(labs) == 0 do	T

Else

F

			if avg_homework < 0.4 || avg_exams < 0.4 || num_labs < 3 do Else cond do #22 mark > 0.895 -> 10 #23 for all options	T F T T

Q 1.4

 Le degré de couverture obtenu est de 91.30%. On a pas réussi à obtenir 100% car il est impossible d'atteindre certaines lignes du code. Par exemple, les figures 1 et 2 montrent des bouts de codes qui ne sont pas atteignables à cause du code qui est présent dans les lignes 3 et 4. Les résultats de la couverture de code sont affichés dans la figure 5. On peut faire face à ces limitation peut être en changeant l'ordre des conditions dans les figures 1 et 2 avec les figures 3 et 4.

```
mark > 0.495 -> "D"          mark > 0.495 -> 2

mark > 0.395 -> "E"          mark > 0.395 -> 1

:else -> "F" #24             :else -> 0 #24
```

```
        "EIN" #19
```

```
if avg_homework < 0.4 || avg_exams < 0.4 || num_labs < 3 do #18
   0 #19
```

Figure 5


```
Finished in 0.2 seconds
16 tests, 0 failures

Randomized with seed 688000

Generating cover results ...

Percentage | Module
-----------|-------------------------
     0.00% | GradesWeb
     0.00% | GradesWeb.ChannelCase
     0.00% | GradesWeb.ErrorHelpers
     0.00% | GradesWeb.PageLive
    50.00% | GradesWeb.LayoutView
    66.67% | GradesWeb.ErrorView
    75.00% | Grades.Application
    75.00% | GradesWeb.Router
    91.30% | Grades.Calculator
   100.00% | Grades
   100.00% | GradesWeb.ConnCase
   100.00% | GradesWeb.Endpoint
   100.00% | GradesWeb.Router.Helpers
   100.00% | GradesWeb.Telemetry
   100.00% | GradesWeb.UserSocket
-----------|-------------------------
    72.29% | Total

Generated HTML coverage results in "cover" directory

D:\2nd year\Semestre 6\SEG3103\Hw\grades\grades>
```

Partie 2

Repo : https://github.com/Hadhoud5/seg3503_playground/tree/main/Assignment_2/Partie 2

Question 2.1 (voir code)

Commits sur Github :
- méthode "avg" ajoutée, "percentage_grade" refractorisé
- "letter_grade" refractorisé
- "numeric_grade" refractorisé

Question 2.2 (voir code)

Commits sur Github :
- Méthode "failed_to_participate?" créée

Question 2.3 (voir code)

Commits sur Github :
- Méthode "calculate_grade" créée

Question 2.4 (voir code)

Commits sur Github :
- 2 refactorisation supplémentaires ajoutés