

CICLO 1 – FUNDAMENTOS DE PROGRAMACIÓN - RETO No. 2

Descripción del problema

Una fundación de carácter social decide otorgar becas para estudio universitario en Colombia y el exterior a estudiantes graduado(a)s de bachilleres en el año 2020, la selección de estudiantes y el tipo beca varía de acuerdo a los siguientes requisitos:

Requisito	Tipo Beca
Promedio de Calificación mayor o igual a 3.5 y estrato menor o igual a 2	La Beca incluye sostenimiento económico y el pago del 100% de los costos académicos.
Promedio de Calificación mayor o igual a 4.0 y estrato igual a 3	La Beca de estudio cubre el 70% de los costos académicos
Promedio de Calificación igual a 3.5 y estrato igual a 3	La Beca de estudio cubre el 40% de los costos académicos
Promedio de Calificación igual o mayor a 4.0 y estrato igual a 4	La Beca de estudio cubre el 30% de los costos académicos

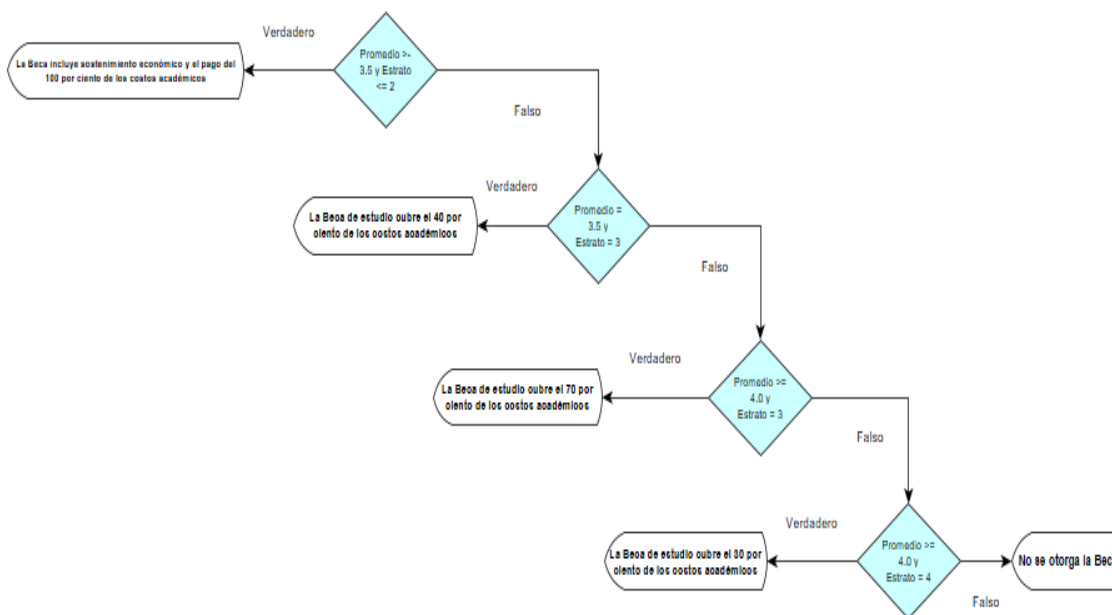


Figura 1 Árbol de decisiones

Se requiere programar dos funciones, una función que reciba la información encapsulada del estudiante en un diccionario, y otra que retorne a partir de un diccionario de respuesta el mensaje de la forma "{NombreEstudiante} – Tipo de Beca: La Beca incluye sostenimiento



económico y el pago del 100 por ciento de los costos académicos.", la beca otorgada al estudiante se determina de acuerdo al árbol de decisiones que muestra la Figura 1.

El esqueleto de la primera función **definir_beca**:

```
def definir_beca(datos_estudiante: dict) -> String:  
    pass
```

Esqueleto de la segunda función **asignación_becas**:

```
def asignación_becas(dicReporte: dict) -> String:  
    pass  
  
# Esta función emite el reporte de la Beca otorgada
```

La fundación cuenta con la siguiente información certificada, la cual viene encapsulada en el diccionario **datos_estudiante**

Entradas

Descripción	Tipo	Descripción
Id_estudiante	str	Código único alfanumérico que identifica al estudiante
nombre_estudiante	str	Estrato Socioeconómico
promedio_calificación	float	Promedio de Calificaciones obtenidas durante el bachillerato
estrato	int	Estrato Socioeconómico

Descripción	Tipo	Descripción
estudiante	str	Nombre del estudiante
Tipo Beca	str	Beca obtenida



Salida

Tipo del Retorno	Estructura	Descripción
dict	diccionario_respuesta = { "estudiante": str "tipoBeca": str }	El diccionario contiene el nombre del estudiante y la beca asignada

Tipo de Retorno	Descripción
str	"{NombreEstudiante} – Tipo de Beca: {Tipo Beca}"

Casos de Prueba

Aquí se presentan algunos ejemplos de como debe responder las funciones solicitadas a unas estradas particulares:

Caso de prueba 1:

```
datos_estudiante = {  
    "id_estudiante": "t001",  
    "estudiante": "Sonia Perez",  
    "promedio": 3.5,  
    "estrato": 2  
}
```

Resultado esperado:

Sonia Perez - Tipo Beca: La Beca incluye sostenimiento económico y el pago del 100 por ciento de los costos académicos.

Caso de prueba 2:

```
datos_estudiante = {  
    "id_estudiante": "t002",  
    "estudiante": "Carmen Soto",  
    "promedio": 4.0,  
    "estrato": 3  
}
```



}

Resultado esperado:

Carmen Soto - Tipo Beca: La Beca de estudio cubre el 70 por ciento de los costos académicos