

PROYECTO SEGUNDO BIMESTRE

Integrantes:

- Juan David Vargas
- David Alejandro Merino

ANÁLISIS

ENTRADA	PROCESO	SALIDA
Ingresar fechas de la inscripción (De inicio y fin)	Verificar si la fecha actual esta en el periodo de inscripción.	
Ingresar nombre de los postulantes	Registrar los datos de los postulantes en arreglos unidimensionales	Presenta en pantalla nombre de postulante
Ingresar carreras elegidas por postulante	Verificar que la carrera ingresada sea válida.	Presenta listado de postulantes registrados
Ingresar puntaje del examen	Verificar si el postulante cumple con el puntaje mínimo de la carrera que haya seleccionado	Presenta puntaje del postulante haya sido admitido o denegado
Ingresar méritos adicionales (solo de medicina)	Calcular el puntaje final de postulantes para medicina sumándolo con los méritos adicionales	Muestra puntaje total de postulantes a medicina
Ingreso de puntaje mínimo para cada carrera	Comparar el puntaje obtenido con el puntaje mínimo de cada carrera seleccionada	Mostrar lista de admitidos para cada carrera con sus puntajes
Ingreso de cupos disponibles por cada carrera	Ordenar los postulantes de las carreras con cupos limitados por puntaje final y méritos, respetando los cupos disponibles	
	Presentar menú con las opciones para registrar postulantes, procesar admisiones o salir del sistema	Mostrar mensaje sobre acción realizada: "Carrera no valida, intente de nuevo".
	Verificar si el puntaje cumple el mínimo establecido para cada carrera	Mostrar mensaje para cada carrera indicando numero de admitidos o no admitidos

DISEÑO

PSEUDOCODIGO

// SISTEMA DE ADMISIONES UNIVERSITARIAS

Definir nombres Como Cadena[100]

Definir carreras Como Cadena[100]

Definir puntajes Como Real[100]

Definir meritos Como Entero[100]

Definir contador_postulantes Como Entero

// Configuración de carreras

Definir lista_carreras Como Cadena[3] = {"Química", "Fisioterapia", "Medicina"}

Definir puntaje_minimo Como Entero[3] = {80, 90, 85}

Definir cupos Como Entero[3] = {80, -1, 80} // -1 significa sin límite

// Función para verificar si las inscripciones están abiertas

Funcion verificar_fecha_actual()

Definir fecha_actual Como Cadena

Escribir "Ingrese la fecha actual (YYYY-MM-DD): "

Leer fecha_actual

Si fecha_actual < "2025-01-20" O fecha_actual > "2025-02-10" Entonces

Escribir "Las inscripciones están cerradas."

Retornar Falso

FinSi

Retornar Verdadero

FinFuncion

// Función para registrar postulantes

Procedimiento registrar_postulante()

Si verificar_fecha_actual() == Falso Entonces

Retornar

FinSi

Definir nombre Como Cadena

Definir carrera Como Cadena

Definir puntaje Como Real

Definir meritos_adicionales Como Entero

Escribir "Ingrese el nombre del postulante:"

Leer nombre

Escribir "Ingrese la carrera (Química, Fisioterapia, Medicina):"

Leer carrera

```

// Validar carrera
Definir indice_carrera Como Entero
indice_carrera = -1
Para i = 0 Hasta 2 Hacer
    Si lista_carreras[i] == carrera Entonces
        indice_carrera = i
    FinSi
FinPara

Si indice_carrera == -1 Entonces
    Escribir "Carrera no válida."
    Retornar
FinSi

Escribir "Ingrese el puntaje obtenido en el examen (0-100):"
Leer puntaje

Si puntaje < 0 O puntaje > 100 Entonces
    Escribir "El puntaje debe estar entre 0 y 100."
    Retornar
FinSi

// Si la carrera es Medicina, pedir méritos adicionales
meritos_adicionales = 0
Si indice_carrera == 2 Entonces
    Escribir "Ingrese méritos adicionales (0-10):"
    Leer meritos_adicionales
    Si meritos_adicionales < 0 O meritos_adicionales > 10 Entonces
        Escribir "Los méritos deben estar entre 0 y 10."
        Retornar
    FinSi
FinSi

// Registrar postulante
nombres[contador_postulantes] = nombre
carreras[contador_postulantes] = carrera
puntajes[contador_postulantes] = puntaje
meritos[contador_postulantes] = meritos_adicionales
contador_postulantes = contador_postulantes + 1

Escribir "Postulante registrado exitosamente."
FinProcedimiento

// Función para procesar admisiones
Procedimiento procesar_admisiones()
    Escribir "--- Resultados de Admisión ---"

    Para i = 0 Hasta 2 Hacer

```

```
Escribir "Carrera: ", lista_carreras[i]
Definir cupos_disponibles Como Entero
cupos_disponibles = cupos[i]
Definir admitidos Como Entero
admitidos = 0
```

```
Para j = 0 Hasta contador_postulantes - 1 Hacer
```

```
  Si carreras[j] == lista_carreras[i] Entonces
```

```
    Definir puntaje_final Como Real
```

```
    puntaje_final = puntajes[j]
```

```
    // Sumar méritos si es Medicina
```

```
    Si i == 2 Entonces
```

```
      puntaje_final = puntaje_final + meritos[j]
```

```
    FinSi
```

```
  Si puntaje_final >= puntaje_minimo[i] Entonces
```

```
    Si cupos_disponibles == -1 O admitidos < cupos_disponibles Entonces
```

```
      Escribir " - ", nombres[j], " | Puntaje: ", puntaje_final
```

```
      admitidos = admitidos + 1
```

```
    FinSi
```

```
  FinSi
```

```
FinSi
```

```
FinPara
```

```
Si admitidos == 0 Entonces
```

```
  Escribir " No hay admitidos para esta carrera."
```

```
FinSi
```

```
FinPara
```

```
FinProcedimiento
```

```
// MENÚ PRINCIPAL
```

```
Proceso principal
```

```
  Definir opcion Como Entero
```

```
  contador_postulantes = 0
```

```
  Repetir
```

```
    Escribir "1. Registrar nuevo postulante"
```

```
    Escribir "2. Procesar admisiones"
```

```
    Escribir "3. Salir"
```

```
    Escribir "Seleccione una opción:"
```

```
    Leer opcion
```

```
  Segun opcion Hacer
```

```
    1:
```

```
      registrar_postulante()
```

```
    2:
```

```
      procesar_admisiones()
```

3:

 Escribir "Saliendo del sistema..."

De Otro Modo:

 Escribir "Opción no válida."

FinSegun

Hasta Que opcion == 3

FinProceso