

# Resolución ejercicios

2023-07-06

## Subsidio desempleo joven

Usted trabaja en una municipalidad y le encargan crear un programa que asigne un subsidio a las personas, en función de su edad y situación en el empleo.

Las reglas son las siguientes: - Edad < 25 y desocupado ==> 125.000 - Edad >= 25 y desocupado ==> 200.000 - Ocupado ==> 0

Los estados de ocupación pueden ser: ocupado o desocupado

En caso de que se cumpla una condición, el programa debe responder lo siguiente: “usted tiene un subsidio de (monto recibido)”

Considere que el programa cuenta con un fondo inicial de 600.000, que va disminuyendo conforme se asignan los recursos. En caso de que el fondo se agote, el programa debe enviar el siguiente mensaje: “ya no quedan recursos”. En caso de que el saldo no alcance a cubrir el subsidio, el programa debe entregar lo que tenga disponible y dejar el fondo en 0.

```
# Parámetros para probar
fondo <- 600000
edad <- 27
situacion_empleo <- "desocupado"

# Si no quedan fondos, el programa no sigue
if (fondo > 0) {

  # Subsidio alto
  if (edad < 25 & situacion_empleo == "desocupado" ) {

    subsidio <- 125000
    diferencia <- fondo - subsidio
    if (diferencia > 0 ) {
      print(paste0("usted tiene un subsidio de ", subsidio))
    } else {
      print(paste0("usted tiene un subsidio de ", fondo ))
    }
  }

  # Subsidio bajo
} else if (edad >= 25 & situacion_empleo == "desocupado") {
  subsidio <- 200000
  diferencia <- fondo - subsidio

  if (diferencia > 0 ) {
    print(paste0("usted tiene un subsidio de ", subsidio))
  }
}
```

```

    } else {
      print(paste0("usted tiene un subsidio de ", fondo ))
    }
    # Sin subsidio
  } else {
    subsidio <- 0
    print(paste0("usted tiene un subsidio de ", subsidio))
  }
  # Actualizar fondo
  fondo <- fondo - subsidio
  print(paste("El saldo es", fondo ))
} else {
  print("ya no quedan recursos")
}

```

```

## [1] "usted tiene un subsidio de 2e+05"
## [1] "El saldo es 4e+05"

```

## 10.000 lanzamientos de moneda

```

cara <- 0
sello = 0
experimentos <- 10000
for (i in 1:experimentos) {
  lanzamiento <- runif(n = 1)
  if (lanzamiento <= 0.5) {
    cara <- cara + 1
  } else {
    sello = sello + 1
  }
}
cara / experimentos

```

```

## [1] 0.4914

```

## Ejemplo adivinanza

```

adivinanza <- 999
numero_secreto <- sample(c(1:10), 1)
print(numero_secreto)

while (numero_secreto != adivinanza) {
  adivinanza <- as.integer(readline("Adivina el número: "))

  if (numero_secreto == adivinanza) {
    print(";acertaste!")
  }
}

```

## Búsqueda binaria

Modifiquemos el juego anterior

1. El rango de valores para buscar será todos los enteros entre 1 a 100
2. Cuando el usuario no acierte, debemos indicar si el número secreto es mayor o menor al número ingresado
3. Debemos guardar los intentos realizados y cuando el usuario adivine, debemos escribir “Felicitaciones, acertaste después de n intentos”
4. Si el usuario escribe “stop”, el programa se detiene y envía el siguiente mensaje: “Gracias por participar en este apasionante juego”

```
adivinanza <- 999
numero_secreto <- sample(c(1:100), 1)
print(numero_secreto)

contar <- 0
while (numero_secreto != adivinanza) {
  # Identificar si el usuario escribe stop
  instruccion <- readline("Adivina el número secreto: ")

  if (instruccion == "stop") {
    print("Gracias por participar en este apasionante juego")
    break
  } else {
    adivinanza <- as.integer(instruccion)
  }

  contar <- contar + 1

  if (numero_secreto == adivinanza) {
    print(paste("¡Felicitaciones! Acertaste después de", contar, "intentos"))
  } else if (numero_secreto > adivinanza) {
    print("El número secreto es mayor")
  } else if (numero_secreto < adivinanza) {
    print("El número secreto es menor")
  }
}
```