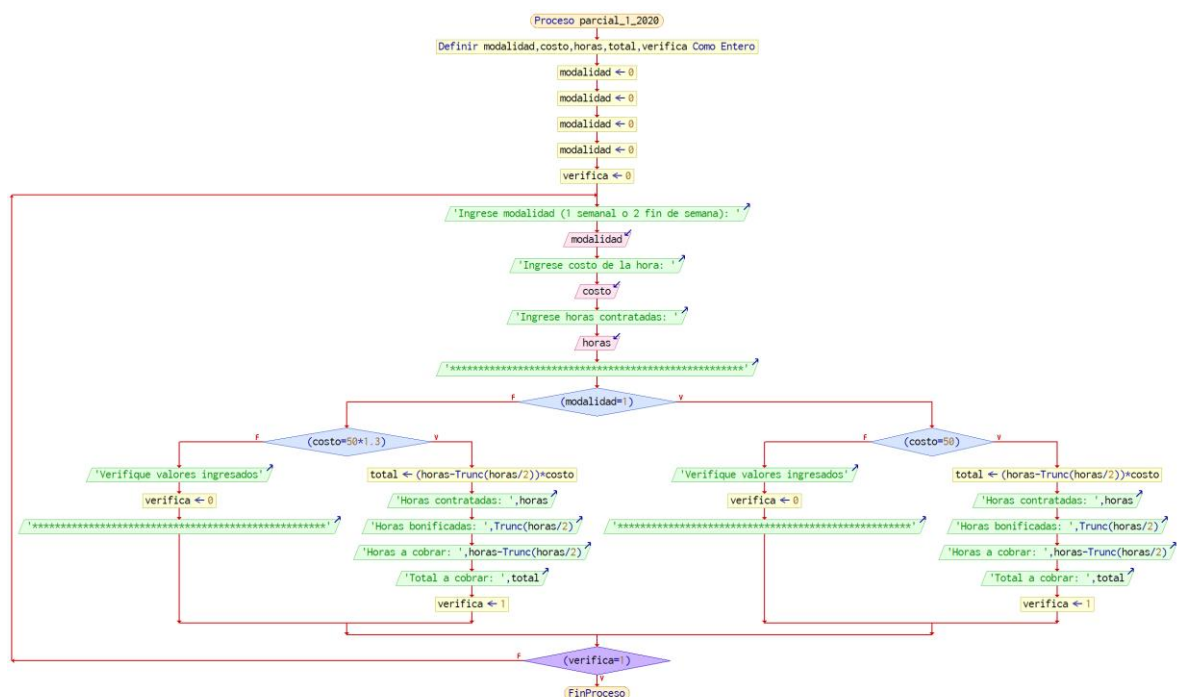


Aspectos procedimentales

Resolver los siguientes enunciados. Para esto se deberá entregar el diagrama de flujos y el código fuente correspondiente a cada uno.

- Calcular el monto a pagar en una cabina de internet si el costo por hora es de \$50 si es durante la semana y 30% adicional si es durante un fin de semana. Para esto el usuario ingresa 1 para la modalidad semanal y 2 para fines de semana. El sistema debe indicar que la opción es incorrecta en caso de ingresar un valor diferente. Por cada hora ingresada, se le bonifica una hora adicional. Se desea saber
 - Cantidad de horas contratadas
 - Monto total de la operación.



```
// Base.
#include <stdio.h>
#include <stdlib.h>

// Internacionalización.
#include <wchar.h>
#include <locale.h>

// Específicas.
#include <math.h>

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

/*
NOTA 1:
Se hizo de esta manera a los fines de usar las estructuras que están siendo
evaluadas en el parcial. Pero el algoritmo es perfectible de ser encarado de una
manera más eficiente.
```

NOTA 2:

El tema de las horas bonificadas merece una consideración. Es práctica bien conocida de que esta modalidad de trabajo no opera con horas fraccionadas. Es decir, si uno deja reservada 1 hora y media, la tarifa corre por las dos horas completas. Por tanto, las bonificaciones (aquí, en este desarrollo) han seguido la misma regla pero balanceándolas a favor del establecimiento comercial. Por ejemplo, en 5 horas contratadas, no se bonificarán ni dos horas y media ni tres horas. Se bonificarán 2.

Por otro lado, el tema de "bonificar 1 hora adicional" puede ser interpretado en tiempo de permanencia o en monto a cobrar. Aquí se hizo por monto a cobrar, pero si la consigna era en tiempo de permanencia, las mismas operaciones permanecen pero la salida debería haber sido así:

```
total = horas * costo;
printf("Horas contratadas:          %d\n", horas);
printf("Horas que puede permanecer: %d\n", horas + (int)floor(horas / 2));
printf("Horas a cobrar:              %d\n", horas);
printf("Total a cobrar:              %d\n", total);
verifica = 1;
```

NOTA 3:

Se podría haber hecho una validación para un ingreso de solo enteros (por ejemplo, verificar que en horas se haya ingresado 7 y no 7.3) pero me pareció que el programa se iba a hacer demasiado extenso y que iba a ser tedioso al profesor. De todas maneras, una verificación de ese tipo se podría hacer así:

```
int num;
char term;

if (scanf("%d%c", &num, &term) != 2 || term != '\n')
    printf("Incorrecto");
else
    printf("Válido");

*/

int main()
{
    // Internacionalización.
    setlocale(LC_ALL, "");

    // Welcome.
    system("color 0A");
    printf("\n");
    printf("Calcular el monto a pagar en una cabina de internet si el costo\n");
    printf("por hora es de $50 si es durante la semana y 30%% adicional si\n");
    printf("es durante un fin de semana. Para esto el usuario ingresa 1 para\n");
    printf("la modalidad semanal y 2 para fines de semana. El sistema debe\n");
    printf("indicar que la opción es incorrecta en caso de ingresar un valor\n");
    printf("diferente. Por cada hora ingresada, se le bonifica una hora\n");
    printf("adicional. Se desea saber:\n");
    printf(" a. Cantidad de horas contratadas\n");
    printf(" b. Monto total de la operación\n");
    printf("\n\n\n");

    // Variables.
    int costo;
    int horas;
    int modalidad;
```

```
int total;
int verifica;
char finale;

// Inicialización.
costo = 0;
horas = 0;
modalidad = 0;
total = 0;
verifica = 0;
finale = 0;

////////////////////////////////////

// HAZ LO TUYO...

// Bucle (herramienta de verificación).
do
{
    // Recupero de datos.
    printf("Ingrese modalidad (1 semanal o 2 fin de semana): ");
    scanf("%d", &modalidad);

    printf("Ingrese costo de la hora: ");
    scanf("%d", &costo);

    printf("Ingrese horas contratadas: ");
    scanf("%d", &horas);

    printf("*****\n");

    // Verificación general de datos ingresados.
    if (horas > 0 && (modalidad == 1 || modalidad == 2) && costo > 0)
    {

        // Validación y total (modalidad 1: días de semana).
        if ((modalidad == 1))
        {
            if ((costo == 50))
            {
                total = (horas - floor(horas / 2)) * costo;
                printf("Horas contratadas: %d\n", horas);
                printf("Horas bonificadas: %d\n", (int)floor(horas / 2));
                printf("Horas a cobrar: %d\n", horas - (int)floor(horas / 2));
                printf("Total a cobrar: %d\n", total);
                verifica = 1;
            }
            else
            {
                printf("Verifique valores ingresados\n");
                verifica = 0;
                printf("*****\n");
            }
        }
    }

    // Validación y total (modalidad 2: fines de semana).
    else
    {
        if ((costo == 50 * 1.3))
```

```
{
    total = (horas - floor(horas / 2)) * costo;
    printf("Horas contratadas:           %d\n", horas);
    printf("Horas bonificadas:           %d\n", (int)floor(horas / 2));
    printf("Horas a cobrar:               %d\n", horas - (int)floor(horas / 2));
    printf("Total a cobrar:               %d\n", total);
    verifica = 1;
}
else
{
    printf("Verifique valores ingresados\n");
    verifica = 0;
    printf("*****\n");
}
}
}
else
{
    printf("Verifique valores ingresados\n");
    verifica = 0;
    printf("*****\n");
}
}

} while ((verifica != 1));

// Goodbye.
// getchar();
printf("\n");
return 0;
}
```