

```

/*
Resolver con ADA el siguiente problema. Se debe modelar el funcionamiento de un
banco donde asisten
N clientes que van a realizar un depósito y que son atendidos por un empleado.
Los clientes pueden ser
Regulares o Premium. Los clientes Regulares solicitan atención y esperan a lo
sumo 1 hora a ser atendidos
y si no se retiran. Los clientes Premium solicitan atención y, si en ese momento
no los pueden atender,
esperan 5 minutos para solicitar atención nuevamente; si luego de 5 intentos no
fueron atendidos, se
retiran del banco. El empleado atiende los pedidos dando prioridad a los
clientes Premium. El cliente le
indica el monto a depositar, y el empleado le devuelve un comprobante de
depósito. Nota: supongo que existe
una función RealizarDeposito(Monto) que realiza el depósito y retorna el
comprobante del mismo.
*/

```

```

procedure Banco is

```

```

task type ClienteRegular;

```

```

task type ClientePremium;

```

```

task Empleado is
    entry EnviarDepositoRegular(monto: in real, comprobante: out text);
    entry EnviarDepositoPremium(monto: in real, comprobante: out text);
end Empleado;

```

```

ClientesRegulares := array (1..R) of ClienteRegular;
ClientesPremium := array (1..P) of ClientePremium;

```

```

task body ClienteRegular is
    monto: real;
    comprobante: text;
begin
    select
        Empleado.EnviarDepositoRegular(monto, comprobante);
    or delay(60*60)
        null;
    end select;
end ClienteRegular;

```

```

task body ClientePremium is
    intentos: integer := 5;
    atendido: bool := false;
    monto: real;
    comprobante: text;
begin
    while ((intentos > 0) and (not atendido)) loop
        select
            Empleado.EnviarDepositoPremium(monto, comprobante);
            atendido := true;
        else
            intentos -= 1;
            delay(60*5);
        end select;
    end loop;
end ClientePremium;

```

```

task body Empleado is
begin
  loop
    select
      accept EnviarDepositoPremium(monto: in real, comprobante: out text)
do
      comprobante := RealizarDeposito(monto);
      end EnviarDepositoPremium;
    or
      when (EnviarDepositoPremium'count = 0) =>
        accept EnviarDepositoRegular(monto: in real, comprobante: out
text) do
          comprobante := RealizarDeposito(monto);
          end EnviarDepositoRegular;
        end select;
    end loop;
end Empleado;

begin
  null;
end Banco;

```