

1 INSTRUCCIONES

- Generar un diagrama de flujo que bosqueje la solución de los siguientes problemas
- Desarrollar un programa que resuelva los siguientes problemas
- Todos los programas deberán subirse a la plataforma GitHub
- Seguir la Rúbricas de Moodle

2 PROBLEMAS

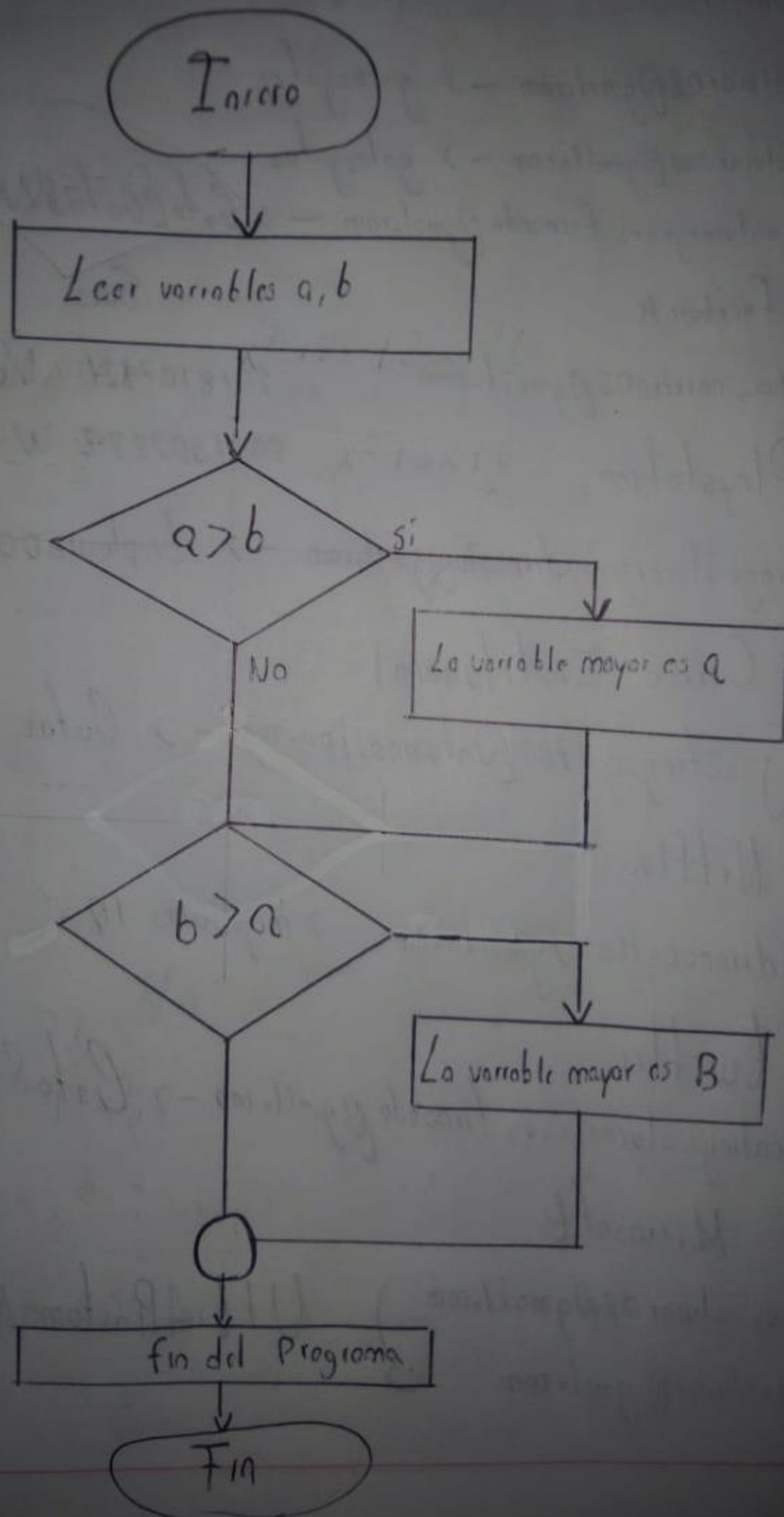
1; Desarrollar un programa que lea los valores a y b, una vez que se hayan leído las variables, el programa debe de indicar cual es la variable mayor.

```
1  #include<stdio.h>
2  int main(){
3      float a;
4      float b;
5      printf("Haremos un programa en cual el valor introducido, dira cual es mayor\n");
6      printf("Dame el valor de a: ");
7      scanf("%f",&a);
8      printf("Dame el valor de b:");
9      scanf("%f",&b);
10     if(a>b)
11     {printf("La variable mayor es a: %.1f\n",a);
12     }
13     if(b>a){
14         printf("La variable mayor es b: %.1f\n",b);
15     }
16     printf("Fin del programa");
17 }
```

hers	
ode::Blocks X Search results X Cccc X Build log X Build messages X CppCheck/Vera++ X	
Line	Message
	=== Build file: "no target" in "no project" (compiler: unknown) ===
	=== Build finished: 0 error(s), 0 warning(s) (0 minute(s), 0 second(s)) ===

Diagrama de flujo Programa 1

FECHA / DIA /



2; Desarrollar un programa que lea una variable tipo entera value, indique si es un valor par o impar.

```
1  #include<stdio.h>
2  #include<math.h>
3  #include<conio.h>
4  int main() {
5      int m;
6      int n;
7      printf("Dame un valor");
8      scanf("%d", &m);
9      n=m%2;
10     if(n==0) {
11         printf("Tu variable es tipo par\n");
12     }
13     else{
14         printf("Tu variable es tipo impar\n" );
15     }
16     }
17     printf("Fin del programa");|
18 }
19
```

others

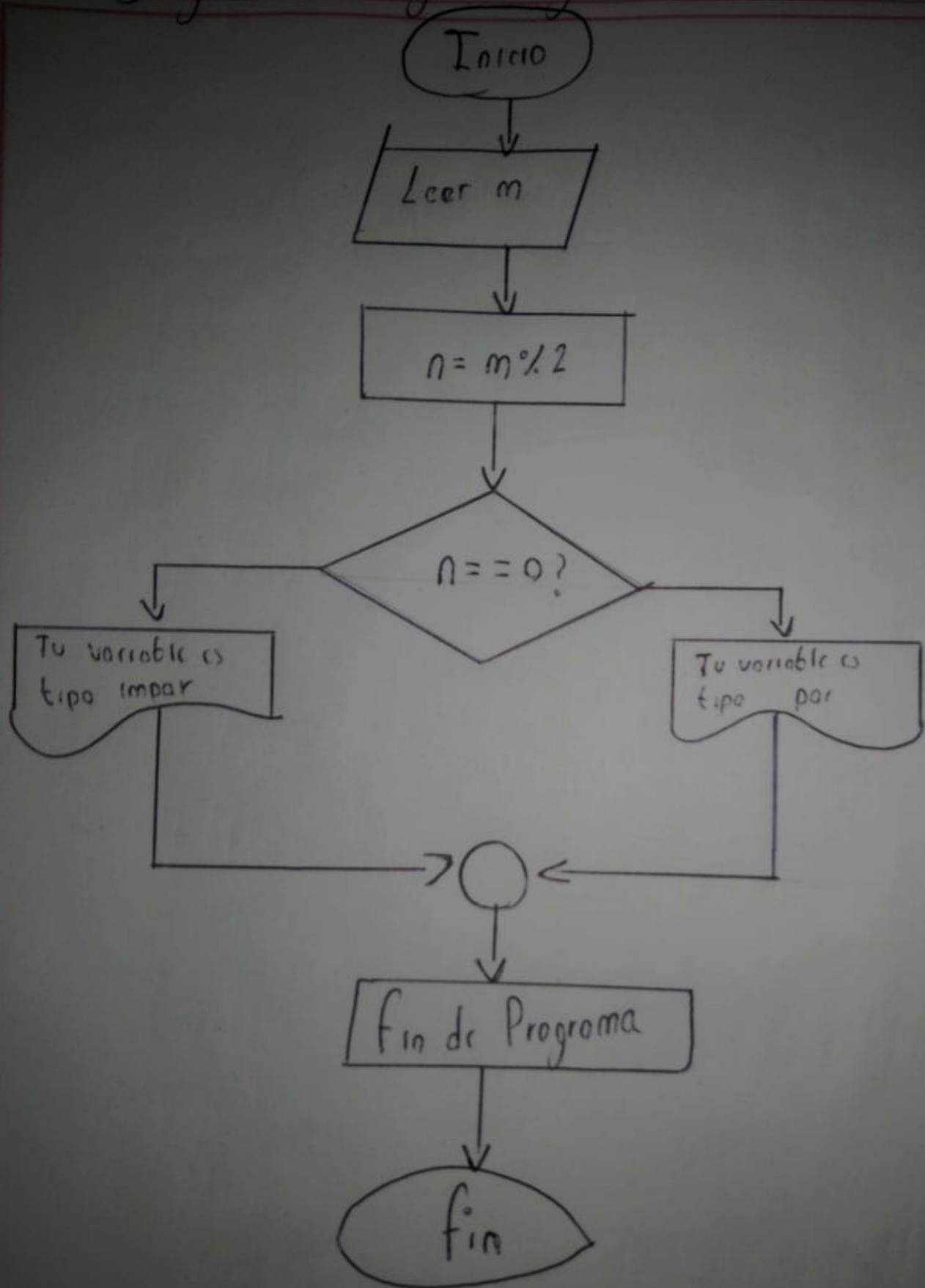
Code::Blocks x Search results x Cccc x Build log x Build messages x CppCheck/Vera++ x

Line Message

=== Build file: "no target" in "no project" (compiler: unknown) ===

=== Build finished: 0 error(s), 0 warning(s) (0 minute(s), 0 second(s)) ===

Diagrama de Flujo Programa 2



4;Suponga que se requieren encontrar las raíces del polinomio de segundo orden

$$Ax^2+bx+c$$

Desarrolle un programa que lea las tres variables y que obtenga el valor de las raíces

$$x1 = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$x2 = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Nota: Para la raíz cuadrada debe de incluir la libreria math.h y puede utilizar la función sqrt(valor). Debe de tener en cuenta que si b^2-4ac debe ser positivo, si no lo es, no debe de ejecutar la operación sino mostrar error

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
int main(){
    float a,b,c,x1,x2,potencia,producto,raiz,cociente;
    printf("Resolveremos una ecuacion de segundo grado\n");
    printf("Ingresa el valor de a\n");
    scanf("%f",&a);
    printf("Ingresa el valor de b\n");
    scanf("%f",&b);
    printf("Ingresa el valor de c\n");
    scanf("%f",&c);
    potencia=pow(b,2);
    producto=(potencia-(4*a*c));
    raiz=sqrt(producto);
    cociente=2*a;
    x1=(-b+(raiz))/cociente;
    x2=(-b-(raiz))/cociente;
    if(raiz>=0){printf("El valor de x1 es: %.2f\n",x1);
    printf("El valor de x2 es: %.2f\n",x2);
    }
    else {printf("No se puede realizar debido a que el discriminante es imaginario");
    }
    if(cociente==0){printf("El Valor es Indeterminado");
    }
    printf("Fin del programa");
}
```

Build log

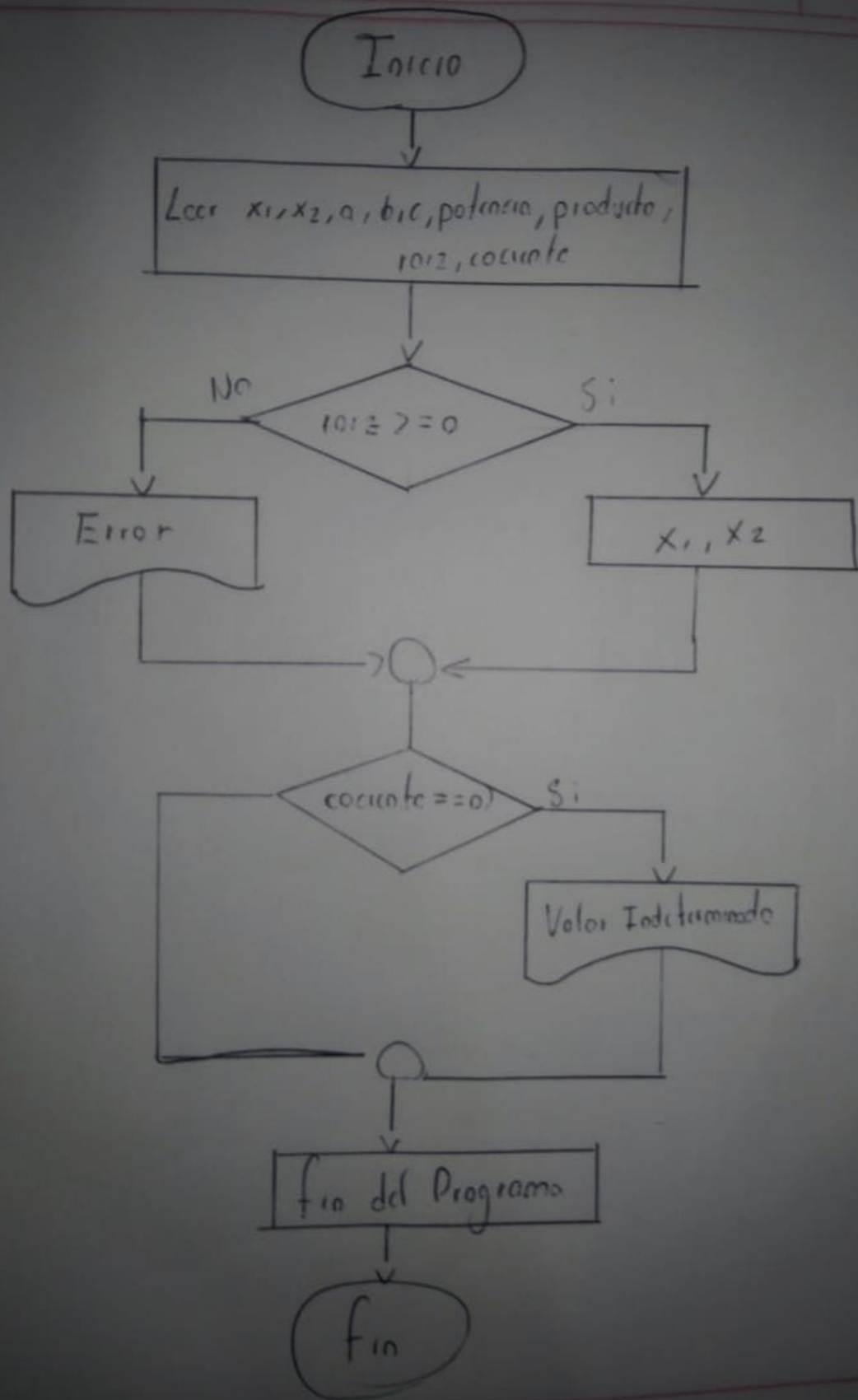
Build messages

CppCheck/Vera++

CppCheck/Vera++ messages

Line	Message
	=== Build file: "no target" in "no project" (compiler: unknown) ===
	=== Build finished: 0 error(s), 0 warning(s) (0 minute(s), 0 second(s)) ===

Diagrama de flujo Program 4

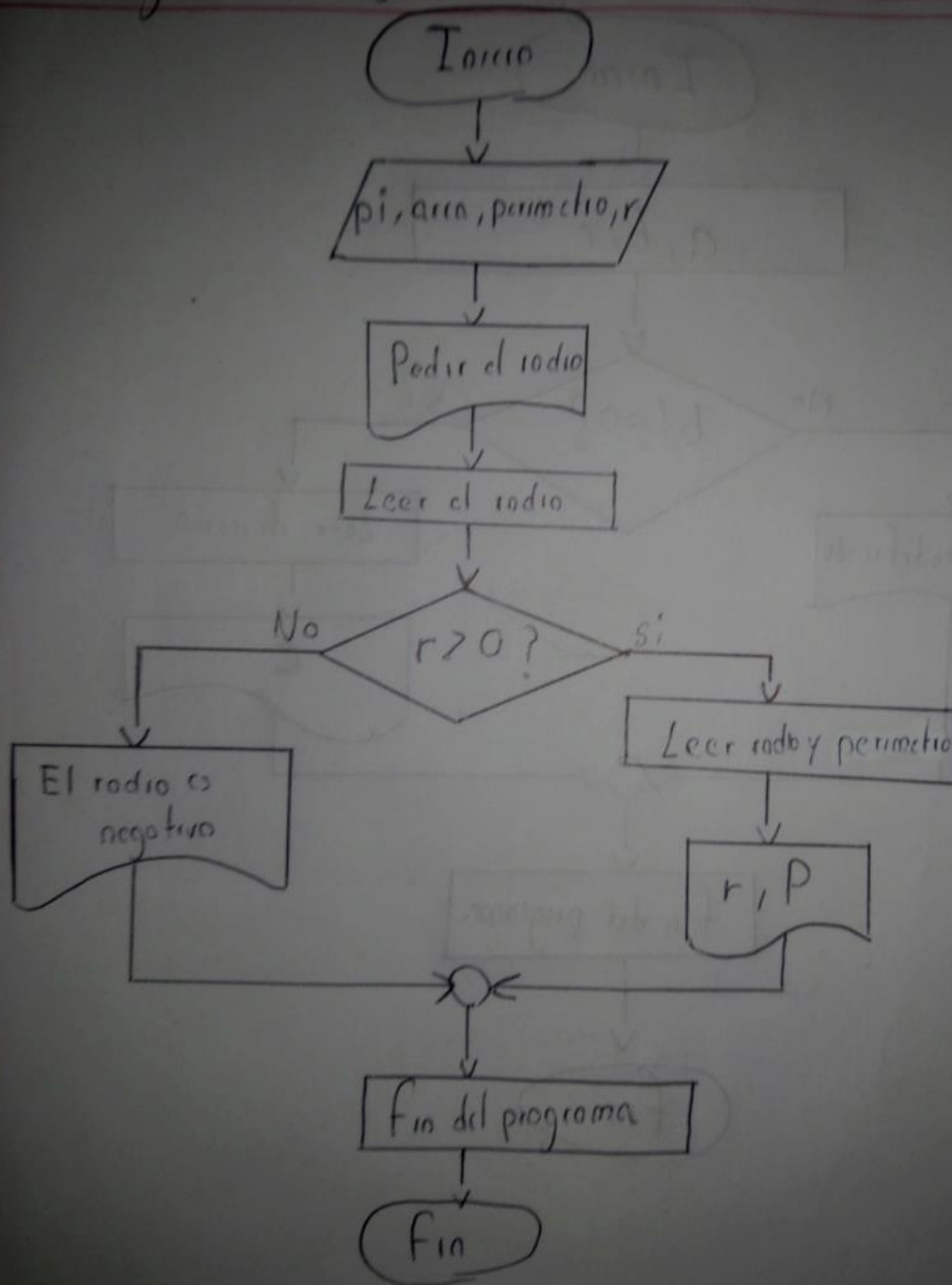


5; Escriba un programa que lea los valores del radio de un círculo y que imprima los valores del área y perímetro.

```
1  #include<stdio.h>
2  #include<math.h>
3  #include<conio.h>
4  int main() {
5      float pi=3.1416;
6      float area;
7      float perimetro;
8      float r;
9      printf("Dame el valor del radio");
0      scanf("%f",&r);
1      if(r>0){area=pow(r,2)*pi;
2          printf("El Area es %f\n",area);
3          perimetro= 2*pi*r;
4          printf("El perimetro es %f\n",perimetro);
5      }
6      else {
7          printf("El radio es negativo, no se puede concluir la operacion\n");
8      }
9      printf("Fin del programa");
0  }
1  }
```

hers	
ccc X	Build log X
Build messages X	CppCheck/Vera++ X
CppCheck/Vera++ messages X	Cscop
Line	Message
	=== Build file: "no target" in "no project" (compiler: unknown) ===
	=== Build finished: 0 error(s), 0 warning(s) (0 minute(s), 0 second(s)) ===

Diagrama de flujo de Programa 5



6; Escriba un programa que lea dos valores y que realice la división a/b , en este caso la variable b no puede ser idéntica a cero, si lo es, debe indicar error y no realizar la operación.

```
1  #include<stdio.h>
2  #include<math.h>
3  int main(){
4      float a;
5      float b;
6      float c;
7      printf("Haremos una division de dos variables\n");
8      printf("Dame el valor primer valor");
9      scanf("%f",&a);
10     printf("Dame el segundo valor");
11     scanf("%f",&b);
12     if(b!=0){c=a/b;
13         printf("El valor del cociente es: %.2f \n",c);
14     }
15     else{printf("Error\n");
16     }
17     printf("Fin del programa");
18 }
19
20
21
```

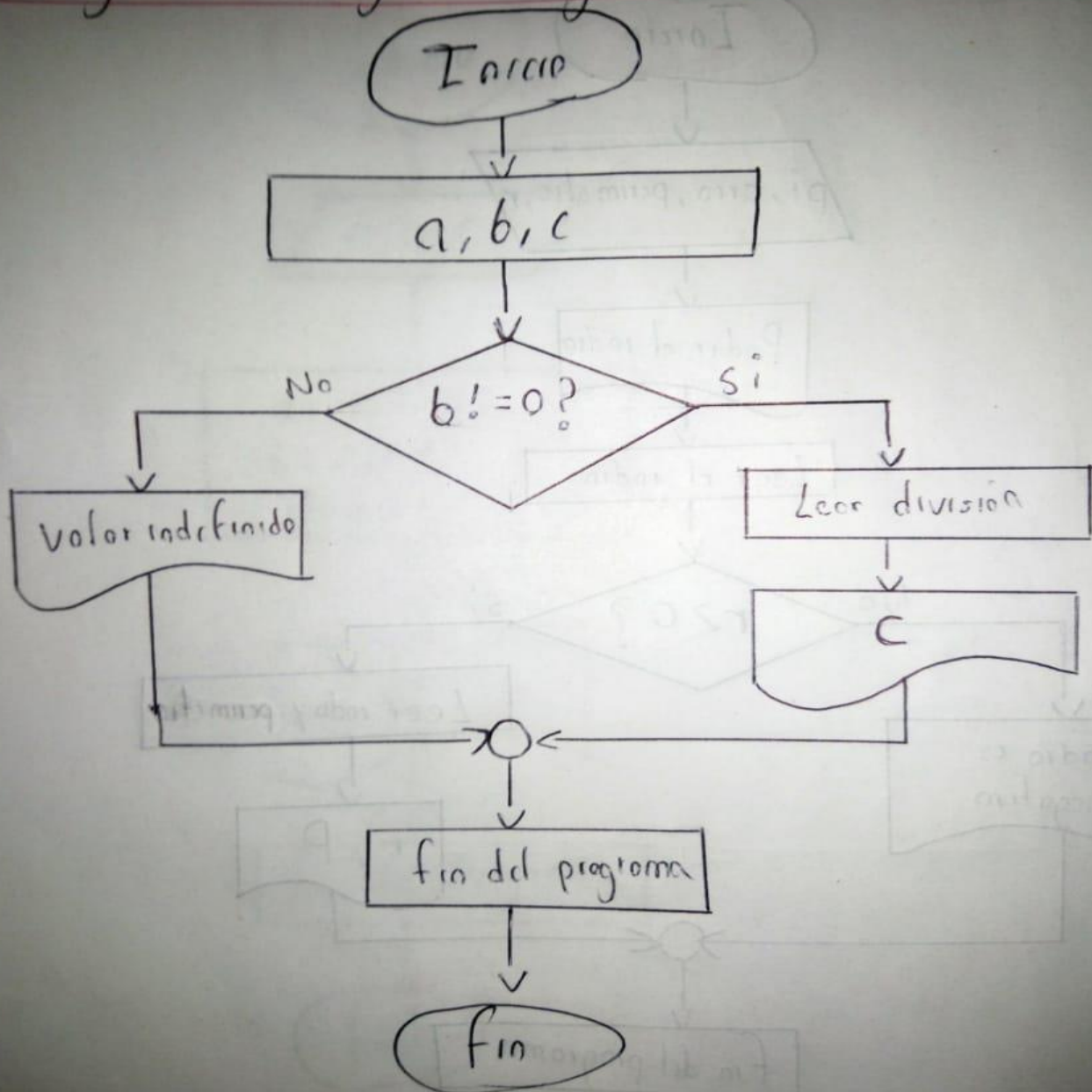
& others

Cccc x Build log x Build messages x CppCheck/Vera++ x CppCheck/Vera++ messages x Cscope

Line	Message
	=== Build file: "no target" in "no project" (compiler: unknown) ===
	=== Build finished: 0 error(s), 0 warning(s) (0 minute(s), 1 second(s)) ===

Diagrama de flujo Programa 6

FECHA / DATE



3 PREGUNTAS

1;Determine cual de los siguientes es un identificador válido

- | | |
|--------------------|----------|
| A. record1 | verde=si |
| B. 1record | rojo=no |
| C. file_3 | |
| D. return | |
| E. \$tax | |
| F. name | |
| G. name and adress | |
| H. name_and_adress | |
| I. name-and-adress | |
| J. 123-45-6789 | |

2;Escriba las declaraciones apropiadas para cada una de las variables

- Variables enteras: **p,q**
Int p;
Int q;
- Variables de tipo flotante: **x,y,z**
float x;
float y;
float z;
- Variables tipo carácter: **a,b,c**
char a;
char b;
char c;

3; Escriba las declaraciones apropiadas para cada una de las siguientes variables

- Variables de punto flotante: **root1**, **root2**
float root1;
float root 2;
- Variables para un entero largo: **counter**
long counter;
- Variable de entera corto: **flag**
short flag;

4;Escriba las declaraciones apropiadas para cada una de las siguientes variables

- Variable entera: **index**

- int index;
- Variable entera sin signo: cust_no
unsigned int cust_no;
- Variable de doble precisión: gros, tax, net
double gros;
double tax;
double net;
- Variable de tipo carácter: current, last
char current;
char last;
- Variable de tipo punto flotante: error
float error;

5;Escriba las declaraciones e inicializaciones de variables, conforme se muestra a continuación

- Variable de tipo flotante: a=-8.2 y b=0.005
 - ✓ float a=-8.2
("%f",a);
 - ✓ float b=0.005;
("%f",b);
- Variable de tipo entero: x=129, y=87 y z=-22
 - ✓ int x=129;
("%d",x);
 - ✓ int y=87;
("%d",y);
 - ✓ int z=-22
("%d",z);
- Variable de tipo carácter: c1='w', c2='&'
 - ✓ char c1='w';
("%c", c1);
 - ✓ char c2='&'
("%c", c2);

6; Explique el objetivo de cada expresión

- a-b resolver una resta de a menos b
- a*(b+c) resolver la suma de b+c y luego multiplicarlo por a
- d=a*(b+c) representar una igualdad
- a>=b representar que a sea mayor o igual a b
- (a%5)==0 representa que el módulo , es decir el residuo de a entre 5 sea completamente igual a 0