



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: Ing. Claudia Rodriguez Espino

Asignatura: Fundamentos de Programación

Grupo: 1102

No de Práctica(s): Práctica número 10

Integrante(s): Alejandro Nava Cruz

Semestre: 2018-1

Fecha de entrega: 12 de Noviembre del 2017

Obervaciones:

CALIFICACIÓN: _____

Practica 10: Depuración de un programa

Objetivo:

Aprender las técnicas básicas de depuración de programas en C para revisar de manera precisa el flujo de ejecución de un programa y el valor de las variables; en su caso, corregir posibles errores.

Desarrollo:

Depurar de cierta manera es ejecutar el programa de una manera más dedicada, sigue el flujo del programa, al igual que con valores pedidos por el usuario y seguir las funciones ya antes puestas (si es que se colocó alguna).

Esta función es muy útil cuando se quiera optimizar el programa, para que al momento de la ejecución se vuelva más fluido, o quieran minimizarse o checarsse errores al momento de compilar, además de ver la manera en la que pueda mejorar un programa, ya sea reduciendo las líneas de código o encontrar métodos alternos para llegar al mismo resultado deseado.

Actividades hechas en casa

1. Ejercicios propuestos por practica:

```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
int N, CONT, AS;
```

```
AS=0;
```

```
CONT=1;
```

```
printf("TECLEA UN NUMERO: ");
```

```
scanf("%i",&N);
```

```
while(CONT<=N)
```

```
{
```

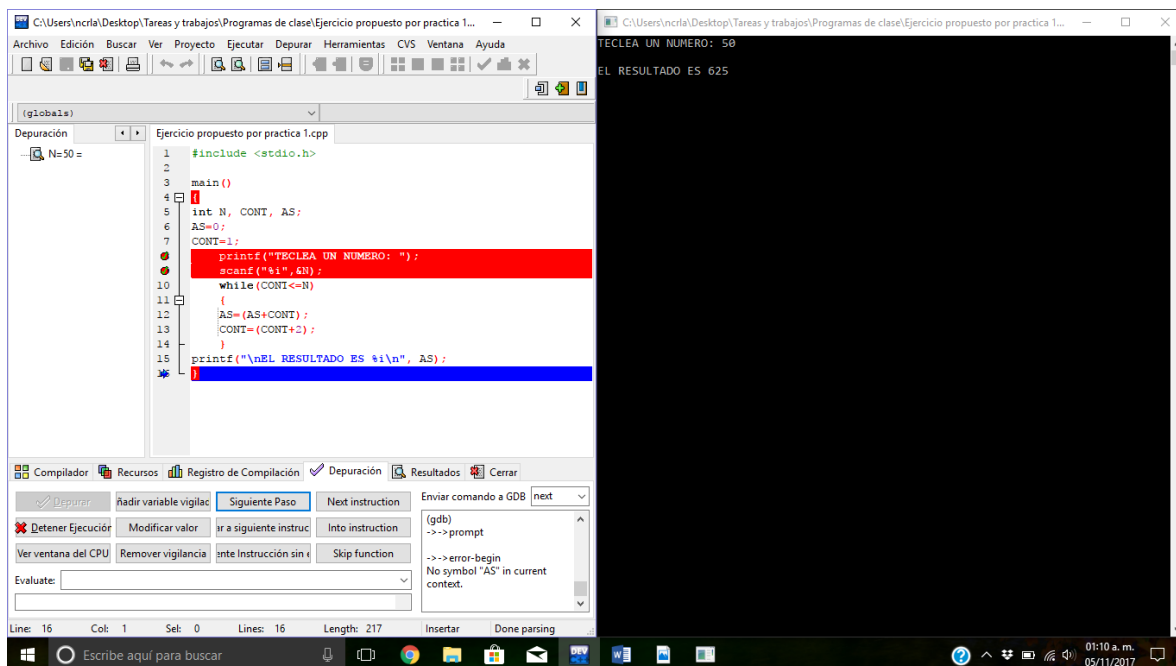
```
AS=(AS+CONT);
```

```
CONT=(CONT+2);
```

```
}
```

```
printf("\nEL RESULTADO ES %i\n", AS);
```

```
}
```



1.2:

```
#include <stdio.h>
```

```
main()
```

```
{
```

```
Int i, j;
```

```
for(i=1; i<10; i++)
```

```
{
```

```
printf("\nTabla del %i\n", i);
```

```
for(j=1; j==10; j++)
```

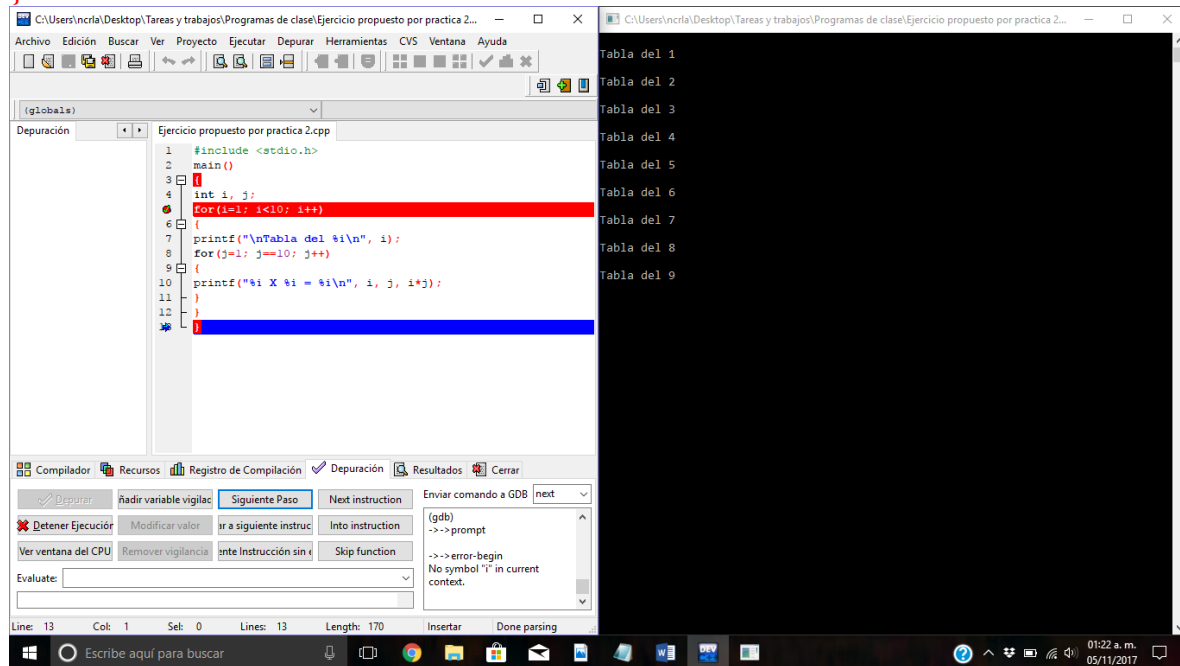
```
{
```

```
printf("%i X %i = %i\n", i, j, i*j);
```

```
}
```

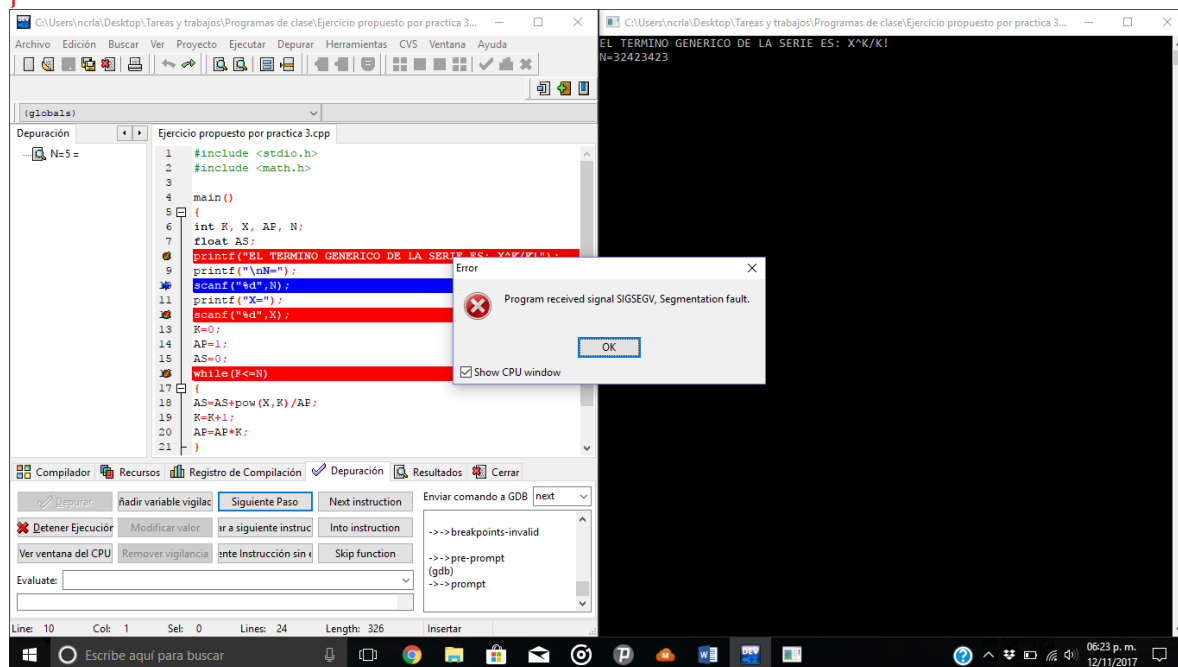
```
}
```

```
}
```



1.3:

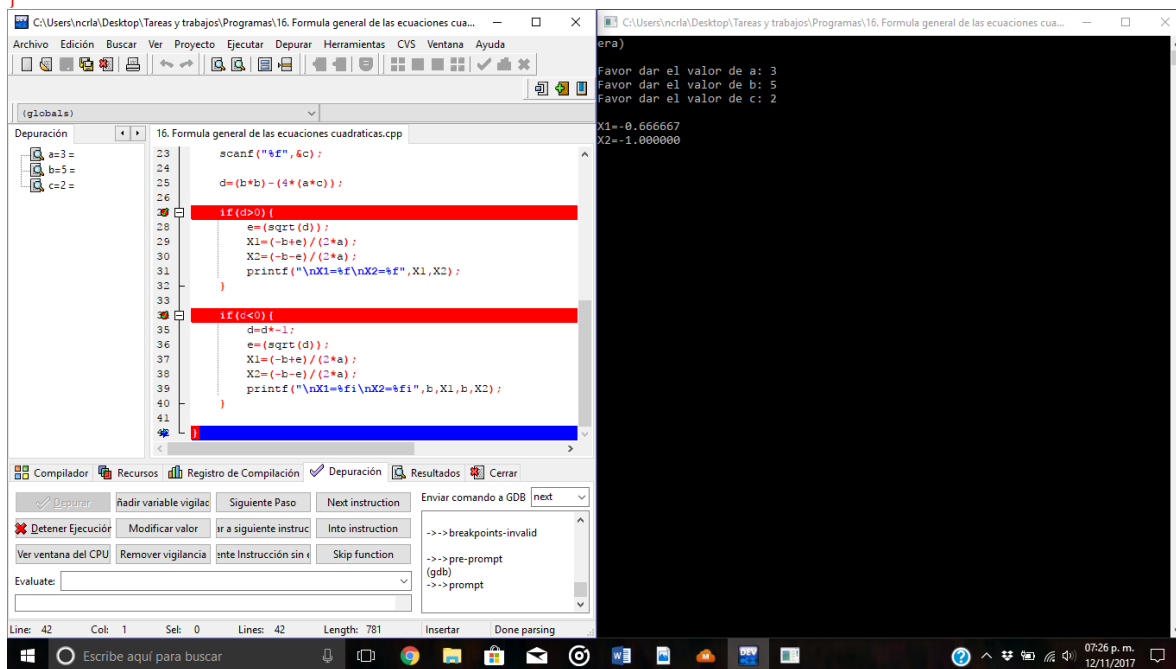
```
#include <stdio.h>
#include <math.h>
main() {
int K, X, AP, N;
float AS;
printf("EL TERMINO GENERICO DE LA SERIE ES: X^K/K!");
printf("\nN=");
scanf("%d",N);
printf("X=");
scanf("%d",X);
K=0;
AP=1;
AS=0;
while(K<=N) {
AS=AS+pow(X,K)/AP;
K=K+1;
AP=AP*K; }
printf("SUM=%le",AS);
}
```



Este programa nos regresa un error gracias a que la variable no se guarda en ningún lugar de memoria, ya que al momento de registrar alguna variable se tiene que guardar con el símbolo "&"

2. Ecuación general de las ecuaciones cuadráticas

```
#include<stdio.h>
#include<math.h>
float a,b,c,x,y,d,e,X1,X2;
main()
{
    printf("\t\tPrograma de resolucio de ecuaciones cuadraticas(Chicharronera)\n\n");
    printf("Favor dar el valor de a: ");
    scanf("%f",&a);
    if(a==0){
        while(a==0){
            printf("\nSi 'a' es igual a 0 se genera una indeterminacion\n");
            printf("Da de nuevo el valor de a: ");
            scanf("%f",&a);
        }
    }
    printf("Favor dar el valor de b: ");
    scanf("%f",&b);
    printf("Favor dar el valor de c: ");
    scanf("%f",&c);
    d=(b*b)-(4*(a*c));
    if(d>0){
        e=(sqrt(d));
        X1=(-b+e)/(2*a);
        X2=(-b-e)/(2*a);
        printf("\nX1=%f\nX2=%f",X1,X2);
    }
    if(d<0){
        d=d*-1;
        e=(sqrt(d));
        X1=(-b+e)/(2*a);
        X2=(-b-e)/(2*a);
        printf("\nX1=%fi\nX2=%fi",b,X1,b,X2);
    }
}
```



Conclusiones:

El depurador es una herramienta muy útil para optimizar programas y para checar y corregir errores de lógica y modificar programa para obtener los resultados esperados

Bibliografía:

Práctica número 10, Facultad de Ingeniería, en Laboratorios A y B (2017), Sitio Web:
<http://lcp02.fib.unam.mx/>