



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Alejandro Pimentel

Profesor:

Fundamentos de programación

Asignatura:

3

Grupo:

8

No de Práctica(s):

Nava Pamatz Oscar Gustavo

Integrante(s):

*No. de Equipo de
cómputo empleado:*

30

35

No. de Lista o Brigada:

2020-1

Semestre:

07/10/2019

Fecha de entrega:

Observaciones:

CALIFICACIÓN: _____

Práctica 8

Objetivos:

Elaborar programas en lenguaje C que incluyan las estructuras de selección if, if-else, switch y ternaria en resolución de problemas básicos.

IF

¿Cómo funciona el if?

```
If (expresión lógica) {  
    // bloque de código a ejecutar  
}
```

“La sentencia if permite controlar qué sentencias de una política se ejecutan mediante la evaluación del valor de una expresión para comprobar si es verdadera. La sentencia if , en el lenguaje de políticas de Impact es la misma que la utilizada en los lenguajes de programación como C/C++ y Java”¹

¿Cómo funciona el SWITCH?

```
switch (opción_a_evaluar){  
    case valor1:  
        /*Código a ejecutar*/  
        break;  
    ...  
    case valorN:  
        /*Código a ejecutar*/  
        break;  
    ...  
    case valorN:  
        /* Código a ejecutar*/  
        break;  
    default:  
        /* Código a ejecutar*/  
}
```

“Switch case es una estructura de control empleada en programación. Se utiliza para agilizar la toma de decisiones múltiples; trabaja de la misma manera que lo harían sucesivos if , if else o until anidados, así como combinaciones propias de determinados lenguajes de programación.”

¿Cómo funciona una condicional?

Condición ? SiSeCumple: SiNoSeCumple;

```
Int a= 3;  
Int b = 4;  
Mayor = a > b ? a:b;
```

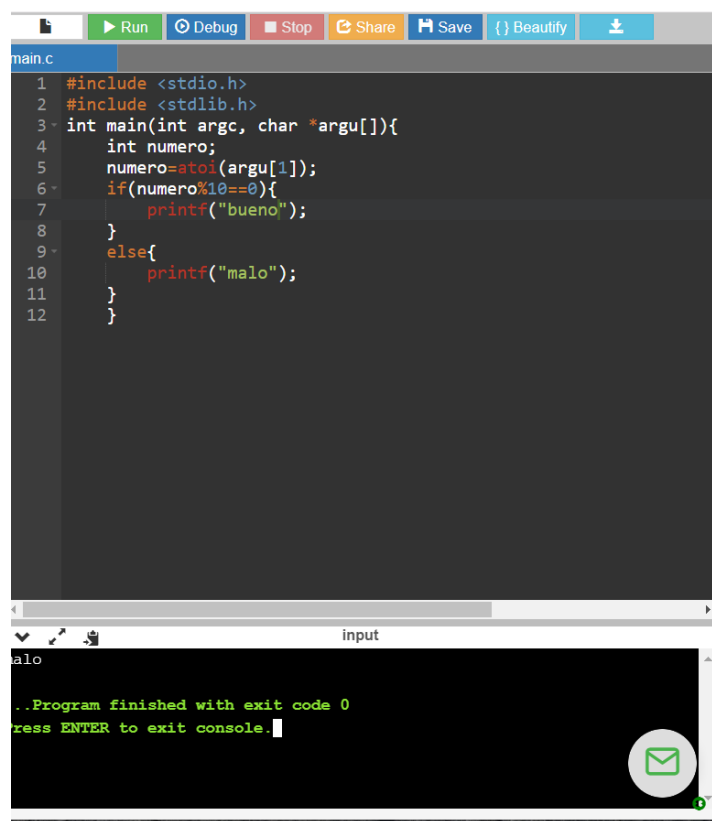
¹ Sentencias if - IBM

https://www.ibm.com › com.ibm.netcoolimpact.doc › policy › if_statements_c

Actividades:

Actividad 1

Experimentar al utilizar diferentes números dentro del condicional del "if" (asegúrense de usar el 0 entre sus elecciones).

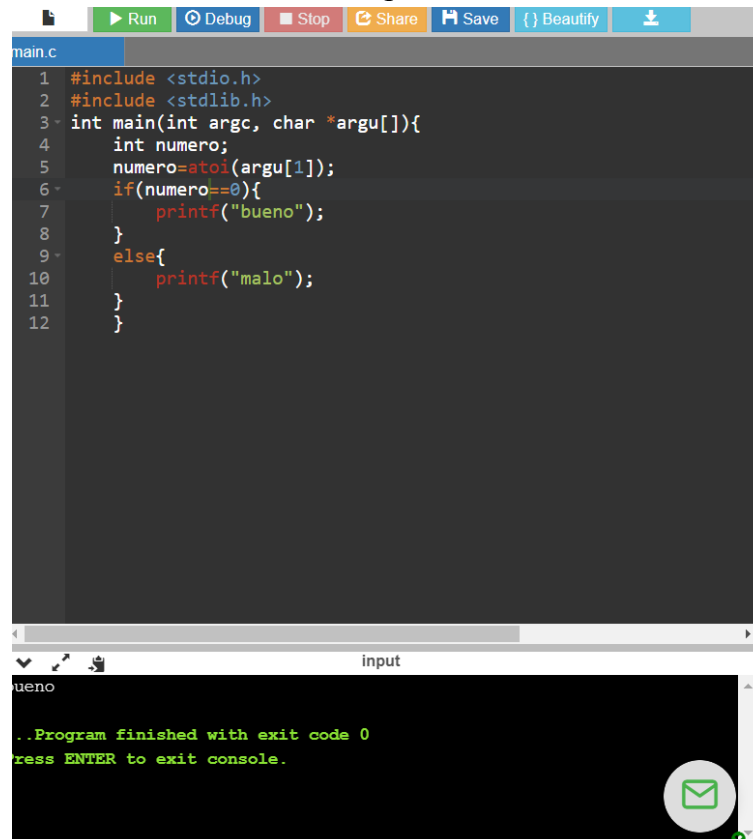


The screenshot shows a C program in a code editor with the following code:

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 int main(int argc, char *argv[]){
4     int numero;
5     numero=atoi(argv[1]);
6     if(numero%10==0){
7         printf("bueno");
8     }
9     else{
10        printf("malo");
11    }
12 }
```

The program is run, and the console output shows "malo" followed by a newline. The console also displays "...Program finished with exit code 0" and "Press ENTER to exit console."

1

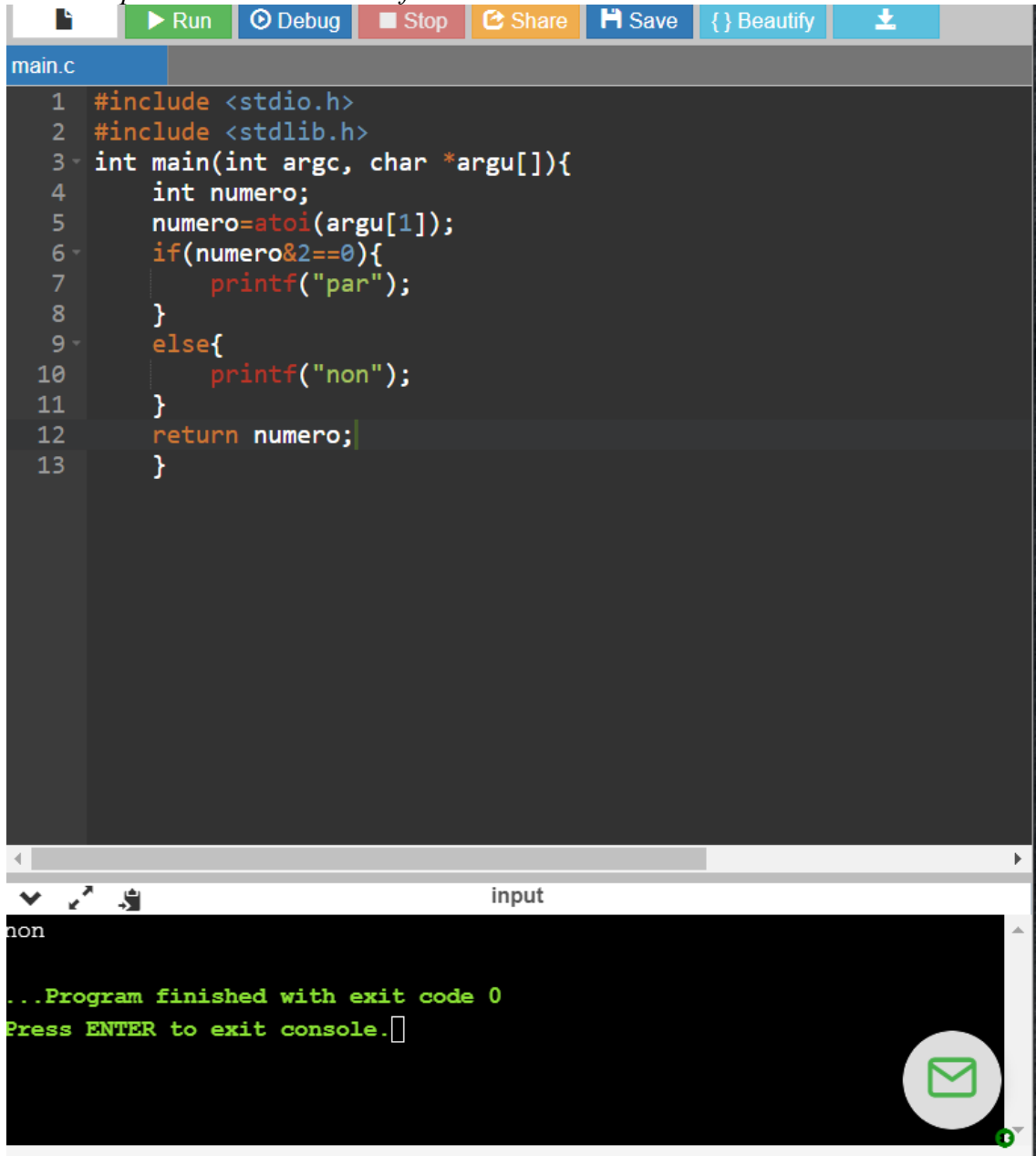


The screenshot shows the same C program as above, but the console output now shows "bueno" followed by a newline. The console also displays "...Program finished with exit code 0" and "Press ENTER to exit console."

Actividad 2

Hacer un programa que lea un número e indique si es par o non.

Nota: para este ocuparé el de la tarea modificando la salida



The image shows a code editor window with a file named 'main.c'. The code is a C program that checks if a number is even or odd. It includes `<stdio.h>` and `<stdlib.h>`. The `main` function takes `argc` and `argu[]` as arguments. It declares an `int numero;`, converts the first command-line argument to an integer using `atoi`, and then checks if the number is even using `numero & 2 == 0`. If even, it prints "par"; otherwise, it prints "non". The program returns the number and ends.

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 int main(int argc, char *argu[]){
4     int numero;
5     numero=atoi(argu[1]);
6     if(numero&2==0){
7         printf("par");
8     }
9     else{
10        printf("non");
11    }
12    return numero;
13 }
```

Below the code editor is a console window titled 'input'. It shows the output 'non' and a message: '...Program finished with exit code 0' and 'Press ENTER to exit console.' with a cursor. There is a green envelope icon in the bottom right corner of the console window.

Actividad 3

Usar condicional para hacer un programa que obtenga el valor absoluto de la diferencia entre dos números.

```
#include <stdio.h>

int main(int argc, char *argv[]){
    float num1;
    float num2;
    float resta;
    float lip;
    num1=atoi(argv[1]);
    num2=atoi(argv[2]);
    resta=num1-num2;
    lip= resta>=0? resta: -resta;
    printf("El valor absoluto es:");
    printf("%f\n",lip);
    return 0;
}
```

input

main.c:8:10: warning: implicit declaration of function 'atoi' [-Wimplicit-function-declaration]
1 valor absoluto es:18.000000

..Program finished with exit code 0
Press ENTER to exit console.

Run Debug Stop Share Save {} Beautify Lang

```
#include <stdio.h>

int main(int argc, char *argv[]){
    float num1;
    float num2;
    float resta;
    float lip;
    num1=atoi(argv[1]);
    num2=atoi(argv[2]);
    resta=num1-num2;
    lip= resta>=0? resta: -resta;
    printf("El valor absoluto es:");
    printf("%f\n",lip);
    return 0;
}
```

input

Command line 1 19

Standard Input: ☒ Interactive Console ☐ Text


```
#include <stdio.h>

int main(int argc, char *argv[]){
    float num1;
    float num2;
    float resta;
    float lip;
    num1=atoi(argv[1]);
    num2=atoi(argv[2]);
    resta=num1-num2;
    lip= resta>0? resta: -resta;
    printf("El valor absoluto es:");
    printf("%f\n",lip);
    return 0;
}
```

input

Command line

Standard Input: ☒ Interactive Console ☐ Text




```
1 #include <stdio.h>
2
3 int main(int argc, char *argv[]){
4     float num1;
5     float num2;
6     float resta;
7     float lip;
8     num1=atoi(argv[1]);
9     num2=atoi(argv[2]);
10    resta=num1-num2;
11    lip= resta>0? resta: -resta;
12    printf("El valor absoluto es:");
13    printf("%f\n",lip);
14    return 0;
15 }
16
17
```

input

main.c:8:10: warning: implicit declaration of function 'atoi' [-Wimplicit-function-declaration]
El valor absoluto es:17.000000

...Program finished with exit code 0
Press ENTER to exit console.



Son dos pruebas con los argumentos deferentes para comprobar si funciona.

Conclusión: podemos concluir que la actividad nos ayuda a entender un poco más del lenguaje c y como utilizar las “palabras claves”, dominando esto podemos hacer muchos programas e ir buscando subir un poco el nivel, sobre todo buscar más funciones y como hacer las cosas de manera más eficiente.

En lo particular me agrado bastante la actividad y si me deja interesado sobre más cosas de programación en lenguaje c.