	Caratula para entrega de prácticas
Facultad de ingeniería	Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: Alejandro Pimentel

Asignatura: Fundamentos de Programación

Grupo: 3

No de Práctica(s): #3

Integrante(s): Arteaga Munguía Erick Alejandro

*No. de Equipo de
cómputo empleado:* Paraguay

No. de Lista o Brigada: 6294

Semestre: 2020-1

Fecha de entrega: 2/08/19

Observaciones:

CALIFICACIÓN: _____

Práctica 3

Objetivo:

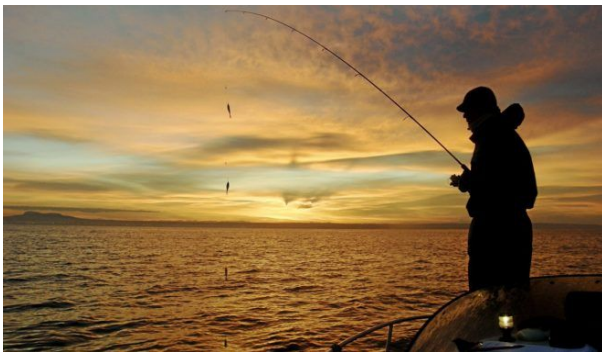
Elaborar algoritmos correctos y eficientes en la solución de problemas siguiendo las etapas de Análisis y Diseño pertenecientes al ciclo de vida del software.

Explicar las precondiciones y el conjunto de salidas de los algoritmos para:

Pescar: Para pescar necesitamos:

1. Una caña de pescar.
2. Una red.
3. Una lancha para poder desplazarse en el agua.
4. Un lago o mar.
5. Asegurarse de que haya peces.
6. Tener carnada para atraer los peces.

La salida de este algoritmo es pescar un buen pescado.



Lavarse las manos:

1. Tener las manos sucias.
2. Tener un baño para lavarse las manos.
3. Tener jabón para desinfectarnos las manos.
4. Asegurarse de que haya un lavabo
5. Tener agua
6. Tener una toalla para secarse las manos

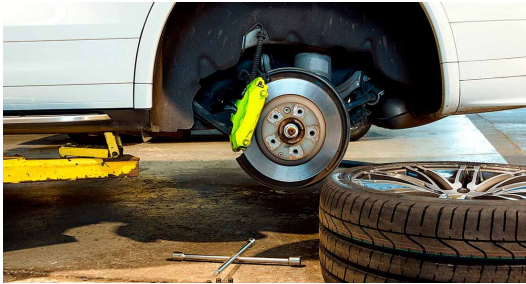
La salida que buscamos es tener unas manos limpias sin bacterias.



Cambiar una llanta:

1. Tener un automóvil.
2. Tener una llanta de refacción
3. Tener la llanta ponchada.
4. Saber cambiar una llanta.
5. Tener las herramientas necesarias para poder cambiar la llanta.

La salida es poder seguir andando con el carro con una buena llanta.



Convertir un numero binario a decimal:

1. Tener un número en binario.
2. Saber convertir los números binarios a decimal.
3. Saber sumar.
4. Saber colocar los exponentes conforme van.
5. Saber los exponentes del dos.

La salida es tener un número en decimal y no en binario.

$$\begin{array}{cccccccc} 64 & 32 & 16 & 8 & 4 & 2 & 1 \\ \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 = \\ 64 + 16 + & 2 & = \\ 82 \end{array}$$

Desarrollar los algoritmos para:

Determinar si un número es positivo o negativo:

1. Examinar el número dado.
2. Observar si tiene un signo.
3. Hacer operaciones para saber cual es el positivo y cual es el negativo.

Obtener el mayor de dos números diferentes:

- 1.- Inicio
- 2.- Declarar: $N1 = \emptyset$; $N2 = \emptyset$
- 3.- Primer número: $N1$
- 4.- Segundo número: $N2$
- 5.- Preguntar: $SI\ N1 > N2$ *Decisión*
- 5.1.- Entonces El N° mayor es: $N1$
- 5.2.- De lo contrario mostrar
- 5.3.- EL N° mayor es: $N2$
- 6.- Fin

Obtener el factorial de un número:

- 1 INICIO
- 2 Define las variables total, contador, numero
- 3 Despliega en la pantalla "Captura el número a determinar su factorial:"
- 4 Captura número
- 5 Si número igual a cero despliega "El factorial es igual a 1", ir a la línea final 14
- 6 Asigna 1 a total
- 7 Asigna a contador el valor de número
- 8 Asigna a total el producto de total por contador
- 9 Asigna a contador la resta de contador menos 1
- 10 Si contador es igual a cero, ir la línea 12
- 11 Ir a línea 08
- 12 Despliega en la pantalla "El factorial del número: ", número
- 13 Despliega en la pantalla "Es igual a:", total
- 14 TÉRMINO

Desarrollar algoritmos propios de un procesador (asignando registros genéricos) para:

Cambiar el signo de un número binario:

1. Agrupar el número binario.
2. Checar el bit de signo.
3. Bajar cada número hasta encontrar el número uno.
4. Cuando encontremos el primer uno lo dejaremos igual pero a partir de él cambiaremos el número.
5. Al terminar de cambiar los numero, verificar si al sumarlos den cero en todo.

Hacer una suma larga binaria:

La tabla de sumar para números binarios es la siguiente:

+
0
1
0
0
1
1
1
1
10

Las posibles combinaciones al sumar dos bits son:

$0 + 0 = 0$
 $0 + 1 = 1$
 $1 + 0 = 1$
 $1 + 1 = 10$

Note que al sumar $1 + 1$ es 10
2

, es decir, llevamos 1 a la siguiente posición de la izquierda (acarreo). Esto es equivalente, en el sistema decimal a sumar $9 + 1$, que da 10: cero en la posición que estamos sumando y un 1 de acarreo a la siguiente posición.

Ejemplo

1
1
0011000
+ 00010101

10101101

Se puede convertir la operación binaria en una operación decimal, resolver la decimal, y después transformar el resultado en un (número) binario. Operamos como en el sistema decimal: comenzamos a sumar desde la derecha, en nuestro ejemplo, $1 + 1 = 10$, entonces escribimos 0 en

la fila del resultado y
llevamos
1 (este "1" se llama
acarreo
o
arrastre
).

A
continuación se suma el acarreo a la siguiente columna: $1 + 0 + 0 = 1$, y seguimos hasta terminar todas las columnas (exactamente como en decimal).

Para concluir aprendimos a formular algoritmos y las precondiciones que necesitamos para cada algoritmo dependiendo de lo que queramos hacer. Además sabiendo las precondiciones podremos predecir cuál serán las salidas de los algoritmos o mejor aún ya la sabemos.

