



Actividad de Aprendizaje Semana 04

Indicaciones Generales:

1. El plagio se sanciona con la suspensión o expulsión del estudiante de la Universidad. Reglamento General de Estudios.
2. Sea cuidadoso con su redacción, la cual formará parte de su calificación (aspectos léxicos, sintácticos, semánticos).
3. Lea bien la pregunta o enunciado antes de proceder a su desarrollo, administre su tiempo eficazmente.

Actividad 1.

1. Leer y escuchar la presentación Semana 4.
2. Descargar el Archivo "Presentación de Actividades.docx", que se encuentra en el ítem Información General.
3. El dicho archivo, resuelva cada ítem de la sección Ejercicios.
4. Guarde el archivo con el nombre: "Act04: Apellidos y Nombres". Ejm. "Act01: Mendoza Cubas Liliana.docx"
5. Adjuntar el archivo a su Canvas, en la tarea "Act04: Ámbito Empresarial".
6. Colocar los resultados en el presente documento e indicar los participantes.

1. Suma y Promedio de números.

```
int s = 0;
double prom =
0;
for (int i = 0; i < n;
    i++) {s = s + A[i];
    prom = s / (double) n;
}
printf("Suma:%d\nProm:%d", s, prom);
```

2. Suma y Promedio de números (2.0).

```
int s = 0;
double prom =
0;
for (int i = 0; i < n;
    i++) {s = s + A[i];
}
prom = s / (double)n;
printf("Suma:%d\nProm:%d", s, prom);
```

3. Ordenamiento de N números del 1 al 100.

```
for (int i = 0; i < n - 1; i++) {
    for (int k = i + 1; k < n;
        k++) {
        if (vec[i] > vec[ k])
        {int aux = vec[i];
        vec[i] = vec[ k];
        vec[ k] = aux;
        }
    }
}
```



```
}
```

4. Ordenamiento de N números del 1 al 100 (2.0).

```
int frec[101] = {0};  
for (int i = 0; i < n; i++)  
    {frec[vec[i]]++;}  
}  
int pos = 0;  
for (int i = 0; i < 101; i++) {  
    for (int k = 0; k < frec[i]; k++) {  
        vec[pos] = i;  
        pos++;  
    }  
}
```

5. Algoritmo raro - Infinito.

```
int i= 0;  
int  
sum=0;  
while (i < 100) {  
    if (i % 2 == 0)  
    {  
        for (int k = 0; k < n; k++)  
            {sum += vec[i];}  
    }  
    else {  
        for (int k=0; k<i; k++)  
            {sum += vec[i];}  
    }  
}
```

6. Factorial.

```
int fact = 1;  
for (int i = 2; i < n; i++)  
    {fact *= i;}  
printf("Factorial: %d", fact);
```

7. Buscar cadena de máximo 50 caracteres.

```
int pos = -1;  
for (int i = 0; i < n; i++) {  
    if (strcmp(vec[i], cadBuscar) == 0)  
    {pos = i;  
    break;}  
}
```

8. Buscar el mayor.

```
int pos = 0;  
for (int i = 1; i < n; i++)  
    {if (vec[i] > vec[pos])  
        pos = i;}  
printf("El mayor es: %d", vec[pos]);
```



9. ¿Otro algoritmo raro – El while es válido?

```
int max = 0;
for (int m = 0; m < n; m++)
{
    int cont = 0;
    int k = m + 1;
    while (vec[m] <= vec[k])
    {
        k = k + 1;
        cont++;
    }
    if (cont > max)
        max = cont;
}
printf("Maximo %d", max);
```

10. Logarítmica.

```
int i=1;
while (i < n) {
    if (vec[i] % 2 == 0)
        i *= 3;
    else
        i *= 2;
}
```

11. Logarítmica (2.0).

```
int i=1;
while (i < n) {
    if (vec[i] % 2 == 0)
        n = n / 3;
    else
        n = n / 2;
}
```

12. Búsqueda binaria – Arreglo ordenado.

```
int inf = 0;
int sup = n - 1;
int pos = -1;

while ((pos == -1) && (sup >= inf)) {
    int medio = (inf + sup) / 2;
    if (arreglo[medio] == 80)
        pos = medio;
    else if (arreglo[medio] < 80)
        inf = medio + 1;
    else
        sup = medio - 1;
}
```



Implemente el código e identifique la notación asintótica, analice las siguientes funciones.

1. Encontrar el número mayor en un arreglo de enteros
2. Ordenar un arreglo de números enteros
3. Eliminar el elemento en una posición de un arreglo
4. Buscar un número en un arreglo
5. Calcular el factorial de N
6. Determinar si un número existe en un arreglo de enteros.
7. Calcular cuántas veces se repite un número X en un arreglo de enteros.
8. Sumar los dígitos de un número entero positivo.
9. Determinar si un número es primo o no.
10. Determinar la cantidad de primos que existen en un arreglo de enteros.