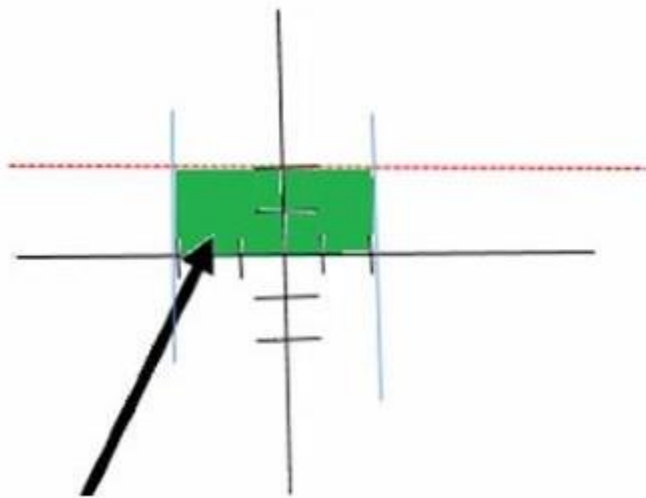
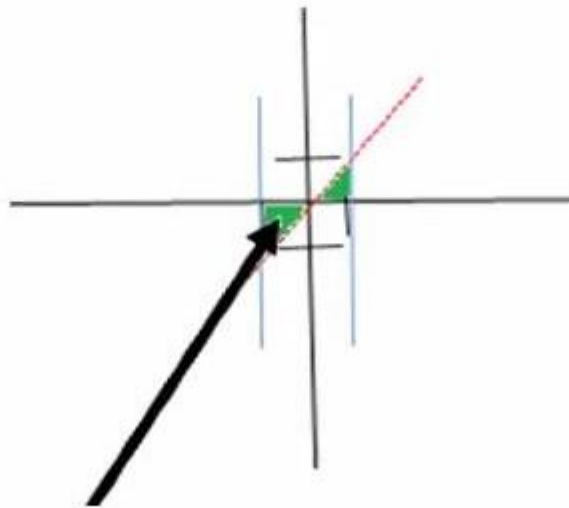


1) IMPLEMENTE LA FUNCIÓN DENOMINADA **AREA (FX: POLINOMIO, A, B: REAL)**

DICHA **FUNCIÓN** DEBERÁ RETORNAR EL VALOR DE **ÁREA** CONSIDERANDO QUE FX ES LA FUNCION Y EL INTERVALO DE CALCULO DEL **ÁREA** SERA A, B CERRADO. (**ÁREA** ES LA ZONA VERDE EN LOS DOS EJEMPLOS **Y DEBE USAR OBLIGATORIAMENTE EL METODO EVALUAR DE LA CLASE POLINOMIO y NO DEBE USAR ALGORITMO DE DERIVADA O INTEGRAL**)

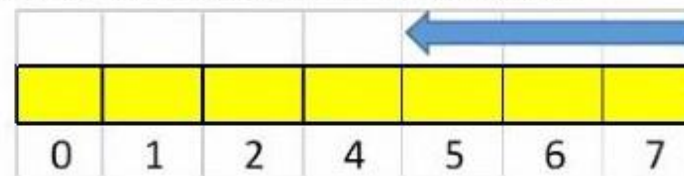


Área = 8 donde $a=-2$ y $b=2$



Área = 1 donde $a=-1$ y $b=1$

- 2) IMPLEMENTE LOS MÉTODOS DENOMINADOS METER(ELEMENTO E) , CREAR Y CIMA DE LA PILA CONSIDERANDO QUE LA PILA ESTA IMPLEMENTADA CON UN ATRIBUTO VECTOR "V???" Y UN OTRO DENOMINADO "TOPE???" DE TIPO ENTERO. ADEMAS QUE LOS DATOS A COLOCARSE EN EL VECTOR INICIAN SIEMPRE EN EL EXTREMO SUPERIOR.



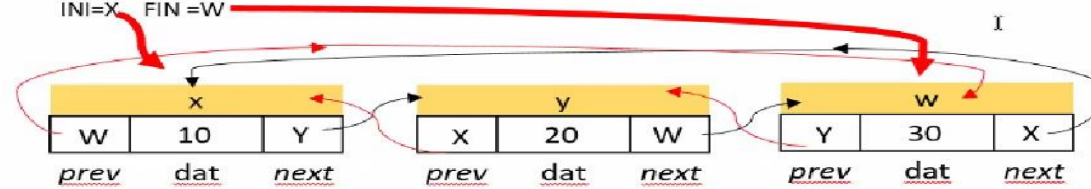
I

Activar Windows
Ve a Configuración para

3) IMPLEMENTE EL PROCEDIMIENTO INVERTIR QUE
TIENE COMO PARAMETRO POR REFERENCIA K???DE
TIPO PILA, EL ALGORITMO DEBE INTERCAMBIAR EL
ULTIMO CON EL PRI MER
ELEMENTO QUE TIENEK???

Eh SI LA P LA TIENE 1.10.100.800ENTOCES EL ALGORITMO
MODIFICAR EL VALOR
DE PARA QUE QUEDE EN 800,10,

4) IMPLEMENTE EL **METODO PONER_FRENTE(ELEMENTO E)** DE LA CLASE DICOLA QUE TIENE COMO ATRIBUTOS INI Y FIN DE TIPO PUNTERO A NODO CON LA ESTRUCTURA (PREV,DAT,NEXT) .
 INI=X FIN=W



Activar Windows
 Ve a Configuración para activar Windows.

Pregunta 4

```
void dicola. poner_frente(int e) {
```

```
Nodo aux = new Nodo;
```

```
if(aux!=Nulo) {
```

```
    aux->dat= e;
```

```
    if(!vacía()) {
```

```
        aux->prev = fin;
```

```
        aux->next = ini;
```

```
        fin->next = aux;
```

```
        ini->prev = aux;
```

```
        ini = aux;
```

```
    }else {
```

```
        aux->next = aux;
```

```
        aux->prev = aux;
```

```
        ini = fin = aux;
```

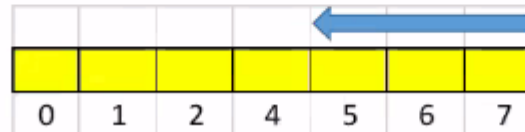
```
}
```

PREGUNTA PARA ESTRUCTURA DE DATOS

- 1) IMPLEMENTE EL METODO DENOMINADO **ESPUNTOSILLA(FI???, CO???)** PERTENECIENTE A LA CLASE CMATRIZDISPERESA IMPLEMENTADA CON VECTORES VD???, VF???, VC??? Y COMPRIMIDA POR FILAS(CRS), SE HACE NOTAR QUE LA FUNCION RETORNA UN VALOR BOOLEANO Y QUE LOS PARAMETROS POR VALOR FI??? Y CO??? DEL METODO SON DE TIPO NUMERICO.
(PUNTO SILLA ES AQUEL ELEMENTO QUE ES MAYOR VALOR EN SU FILA Y MENOR EN SU COLUMNA)

PREGUNTA PARA ESTRUCTURA DE DATOS.

2) IMPLEMENTE LOS **MÉTODOS** DENOMINADOS **METER(ELEMENTO E)** , **CREAR Y CIMA** DE LA PILA CONSIDERANDO QUE LA PILA ESTA IMPLEMENTADA CON UN ATRIBUTO VECTOR "V???" Y UN OTRO DENOMINADO "TOPE???" DE TIPO ENTERO. ADEMAS QUE LOS DATOS A COLOCARSE EN EL VECTOR INICIAN SIEMPRE EN EL EXTREMO SUPERIOR.



PREGUNTA 3:

```
void invertir(pila* KRDMO)
{
    pila *aux= KRDMO;
    pila *final=new pila();
    pila *primera= new pila();
    pila *aux2=KRDMO;
    int num1;
    int num2;
    num1=aux->cima();
    while(!aux2->vacía())
    {
```

```
    int num;

    aux2->sacar(num);

    primera->meter(num);

    if(aux2->vacía())

    {

        num2=num;

    }

}
```

```
while(!primera->vacía())
```

```
{

    if(final->vacía())

    {

        final->meter(num1);

        int numero;

        primera->sacar(numero);

    }

    else

    {

        int numeroAux;

        primera->sacar(numeroAux);

        if(primera->vacía())

        {
```



```
        final->meter(num2);

    }

    else

    {

        final->meter(numeroAux);

    }

}

}

KRDMO=final;
```

```
}
```

```
void invertir(Pila* KEOS){

    Pila* aux = new Pila();

    int a,b,E ;

    KEOS->Sacar(a);

    while (!KEOS->Vacia()){

        KEOS->Sacar(E);

        aux->Meter(E);

    }

    aux->Sacar(b);
```

```
KEOS->Meter(a);

while (!aux->Vacia()){

    aux->Sacar(E);

    KEOs->Meter(E);

}

KEOS->Meter(b);

}
```

```
void Dicola::poner_frente(int e) {
```

```
    NodoC* aux = new NodoC();

    aux->dat = e;

    if (vacía()) {

        aux->prev =aux;

        aux->next =aux;

        ini = aux;

        fin = aux;

    }

    else {

        aux->next =ini;

        ini->prev =aux;

        aux->prev =fin;
```

```
fin->next =aux;
```

```
ini = aux;
```

```
}
```

```
}
```