

Programación competitiva en la UNAL

2020 - 02

Contenido

1. Problemas de programación
2. Competencias de programación
3. La UNAL en las competencias
4. Relación a nivel empresarial
5. Recomendaciones y entrenamiento
6. Grupo de programación competitiva UNAL 2020 - 02
7. Contacto

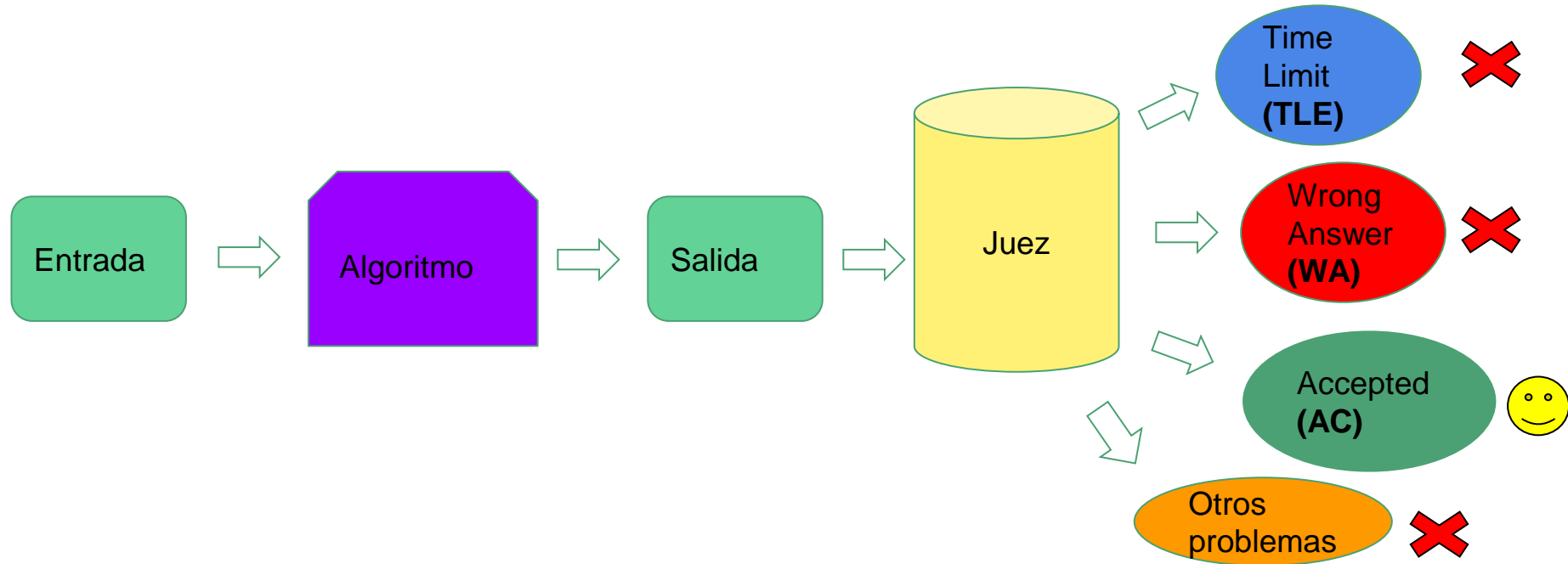
Problemas de programación

¿Qué es un problema de programación?

Es un **problema** (matemático, de grafos, de simulación, etc) que se puede/quiere resolver con un computador a través de algún programa en un **lenguaje de programación**

- ¿Qué se quiere resolver?
- Variables del problema (límites)

Problema tradicional



Ejemplo: $A + B$

Dados 2 números A y B imprimir $A + B$.

Entrada: 2 números (A y B)

Salida: La suma de los números

Casos de ejemplo	Entrada	Salida
Ejemplo 1	5 7	12
Ejemplo 2	3 8	11

Posibles soluciones: A + B

```
#include<iostream>
using namespace std;
int main(){
    int a,b;
    cin>>a>>b;
    cout<<(a+b)<<endl;
}
```

```
#include<iostream>
using namespace std;
int main(){
    double a,b;
    cin>>a>>b;
    cout<<setprecision(5)<<(a+b)<<endl;
}
```

```
#include<iostream>
using namespace std;
int main(){
    int a,b;
    cin>>a>>b;
    for(int i =0;i<b;i++)a++;
    cout<<a<<endl;
}
```

Pregunta:

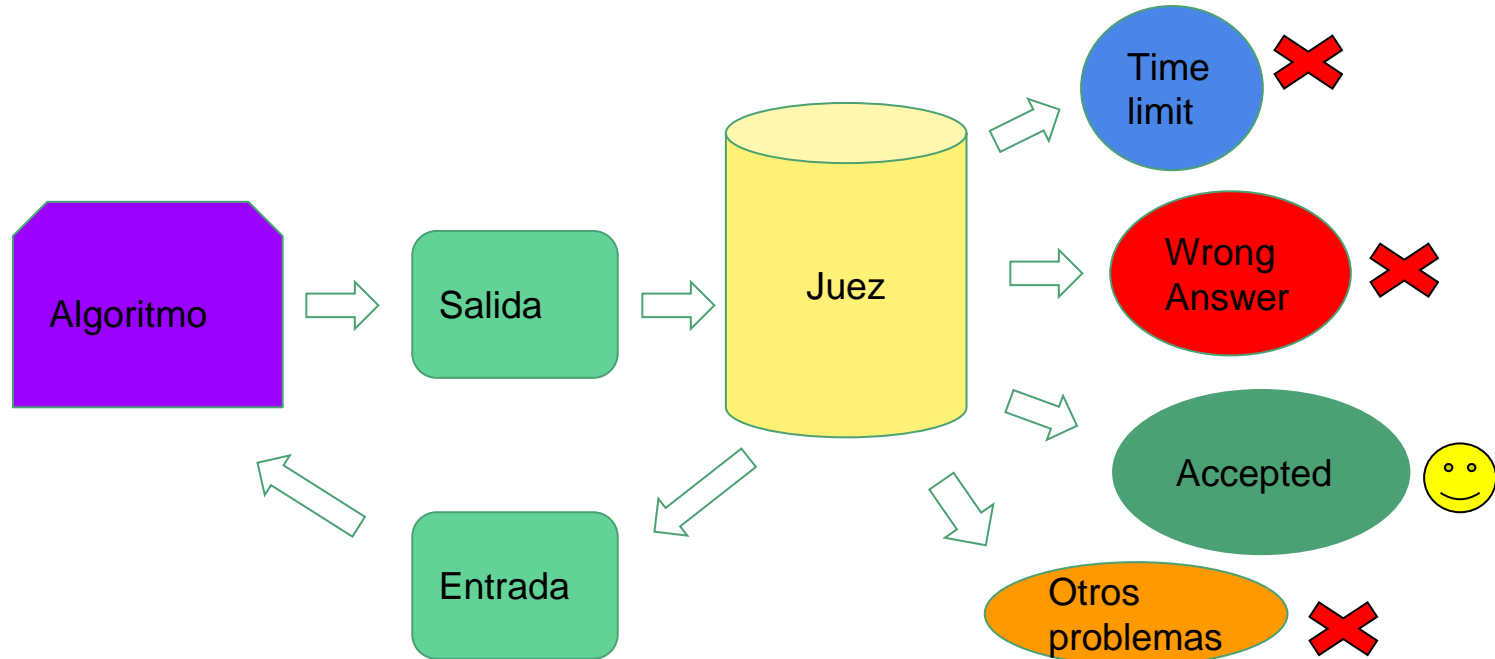
¿Qué diferencias hay entre estas soluciones? ¿sirven?

Restricciones: $A + B$

- ¿Son A y B números enteros o con punto decimal?
- ¿Son A y B siempre positivos o pueden ser negativos?
- ¿Que tan grandes puede ser A y B?

Las restricciones del problema son parte del problema y por ende (usualmente) están estipuladas dentro del problema

Problema interactivo



Ejemplo: Adivina el número

El juez tiene un número entero X ($0 < X < 10^{15}$) usted debe tratar de adivinar el valor de X , al decirle un número Y al juez este le responderá si este es mayor, menor o igual

Interacción: usted imprime un número Y , su predicción sobre el número X

el juez envía “ARRIBA”, “ABAJO” o “IGUAL” si Y es mayor, menor o igual, respectivamente, a X

Casos de ejemplo	Salida	Entrada
Ejemplo 1 ($X = 2$)	5 3 1 2	ABAJO ABAJO ARRIBA IGUAL

Posibles soluciones: Adivina el número

```
#include<iostream>
using namespace std;
int main(){
    long long y = 500000000000000011;
    string veredicto;
    cout<<y<<endl;
    cout.flush();
    cin>>veredicto;
    while(veredicto != "IGUAL"){
        if(veredicto == "ARRIBA")y++;
        else y--;
        cout<<y<<endl;
        cout.flush();
        cin>>veredicto;
    }
}
```

```
#include<iostream>
using namespace std;
int main(){
    long long low = 1, high = 1000000000000000011, mid;
    string veredicto;
    while(low < high){
        mid = (low + high)/2;
        cout<<mid<<endl;
        cout.flush();
        cin>>veredicto;
        if(veredicto == "IGUAL")break;
        if(veredicto == "ARRIBA")low = mid + 1;
        else high = mid - 1;
    }
}
```

Tipos de problemas

1. Simulaciones de ciertos escenarios
2. Problemas de conteo y combinatoria
3. Problemas de optimización discreta
4. Problemas de grafos
5. Geometría computacional
6. Análisis y manipulación de cadenas de texto
7. ...

En general problemas que requieran de conocimientos de matemáticas y algoritmos para ser resueltos de manera **eficiente**.

Asignaturas relacionadas

- Programación de computadores, Programacion y metodos numericos
- Programación orientada a objetos, Estructuras de datos
- Algoritmos, Introducción a las ciencias de la computación
- Geometría
- Matemáticas discretas 1 y 2, Fundamentos de matemáticas, Sistemas numéricos, Grupos y anillos
- Teoria de numeros, Teoria de grafos, Teoría de la computación
- Probabilidad y estadística
- Álgebra lineal, Cálculos, Métodos numéricos,
- Optimización, Redes de flujos

Competencias de programación

¿Cómo funciona una competencia de programación?

Se da un conjunto de problemas de programación que se deben tratar de resolver en un tiempo determinado, el ganador es quien haya resuelto la mayor cantidad de problemas en el menor tiempo

- Competencias Virtuales:

- Codejam
- Hackercup
- IEEEExtreme
- ...

- Competencias Presenciales:

- ICPC
- ...

ICPC

La ICPC (International Collegiate Programming Contest) es la competencia de programación más reconocida a nivel mundial.

En esta varios equipos universitarios compiten para probar cual de ellos puede resolver la mayor cantidad de problemas con el menor tiempo.

Existen varias competencias que tienen este formato:

- Equipos de 3 personas
- Set de 10+ problemas
- Duración 5 horas
- Penalidad de 20 minutos por envío incorrecto
- Sin acceso a material electrónico

Elegibilidad (2019 - 2020)

Las competencias regionales se hacen en el segundo semestre de cada año y la competencia mundial en el primer semestre del año inmediatamente posterior

<https://icpc.baylor.edu/regionals/rules>

- Participar a nombre de una sola institución de educación superior
- No haber competido en más de 5 regionales ni 2 mundiales

Cumplir al menos alguna de las siguientes:

- Haber iniciado en la universidad en 2015
- Haber nacido en 1996 o después
- NO haber completado mas de 8 semestres de estudio en tiempo completo

Competencias virtuales

- IEEExtreme (<https://ieeextreme.org/>)
- Codejam (<https://codingcompetitions.withgoogle.com/codejam>)
- Hackercup (https://www.facebook.com/hackercup/past_rounds/0/)
- Codeforces (<https://codeforces.com/>)
- Topcoder (<https://arena.topcoder.com/>)
- Hackerearth (<https://www.hackerearth.com/challenges/>)
- UVA (<https://uva.onlinejudge.org/>)
- Kattis (<https://open.kattis.com/>)
- ...

La UNAL en las competencias

2006



San Antonio, USA

2009



Estocolmo, Suecia

2010 - 2011



Orlando, USA

2011 - 2012



Varsovia, Polonia

2012 - 2013



St Petersburg, Russia

2013 - 2014



Ekaterimburgo, Rusia

2014 - 2015



Marrakesh, Marruecos

2015 - 2016



Phuket, Thailand

2016 - 2017



Rapid City, USA

2017 - 2018



Beijing, China

2018 - 2019



Porto, Portugal

2019 - 2020



<https://acis.org.co/portal/content/comunicado-marat%C3%B3n-mundial-icpc-2020>

Moscú, Rusia

Relación a nivel empresarial

Competencias Desarrolladas

- Adaptar problemas de la vida real a problemas en términos conceptuales
- Capacidad para solucionar problemas algorítmicos/matemáticos
- Mejorar habilidades de programación
- Optimizar el manejo de tiempo
- Identificar que es un problema fácil/difícil
- Organización y trabajo en equipo
- Lectura en inglés
- ...

Oportunidades

Empresas de Software:

- Microsoft
- Google
- Facebook
- Amazon
- ...

Muchos (ex)competidores han sido pasantes y/o son actualmente desarrolladores de tiempo completo en empresas líderes en tecnología en computación a nivel mundial.

Recomendaciones y entrenamiento

Recomendaciones



Recomendaciones

- Iniciar lo más pronto posible (elegibilidad)
- Estudiar teoría de los algoritmos clásicos
- Solucionar muchos problemas
- Asistir a cualquier competencia presencial / virtual que puedan
- Asistir a los entrenamientos que ofrecemos
- No todos son AC, no achantarse por eso
- Hacer upsolving de los problemas que hayan fallado
- Ser paciente y no rendirse, llegar a una mundial puede tomar años
- Entrenar, esto es como un deporte
- Organizar bien el tiempo y las asignaturas
- Las vacaciones son un buen momento para mejorar y aprender mucho
- No están solos, pidan ayuda a personas mas experimentadas
- **Divertirse y aprender**

Recursos Virtuales

- <https://www.coursera.org/specializations/data-structures-algorithms>
- <http://codeforces.com/blog/entry/13529>
- <https://www.topcoder.com/community/data-science/data-science-tutorials/>
- <https://www.hackerearth.com/practice/codemonk/>
- http://www.comp.nus.edu.sg/~stevenha/myteaching/competitive_programming/cp1.pdf

Organización 2020 - 01

Organización por niveles

No todas las personas están en el mismo nivel

- Básica: Personas que apenas tengan conceptos basicos de programacion
- Media: Conocimientos de estructuras de datos y algo de práctica en competencias
- Avanzada: Equipos que ya manejan bien los algoritmos y temas clásicos de la programación competitiva, que ya estén en la capacidad de entrenarse para representar a la UNAL a nivel regional y mundial

Se realizan entrenamientos acordes a estos niveles los viernes a las 2 pm en el salon 102 de Aulas de Ingeniería (edificio 401). En el grupo de facebook se avisa el tema a entrenar y el nivel

Competencias oficiales 2020

- Maratón Interna UNAL: ? (probablemente inicios de julio)
- Maratón Nacional de Programación ACIS-Redis: 21 de septiembre
- Maratón Regional Latinoamericana ICPC-ACIS-Redis: A mediados de noviembre

3 equipos permitidos por carrera, sujetos a elegibilidad

Competencias oficiales 2019

CCPL <https://www.programmingleague.org:>

- Las CCPL son competencias que simulan el ambiente real de una competencia oficial ICPC, funcionan 5 horas por la tarde los sábados en equipos de 3 personas. Estas competencias se realizan en diferentes universidades que hacen parte de la liga de competencias de programación

RPC [http://redprogramacioncompetitiva.com/:](http://redprogramacioncompetitiva.com/)

- Las RPC son competencias que se hacen en simultáneo en toda latinoamérica, en Bogotá se puede participar de manera virtual

Contacto

Contacto

Site del grupo <https://sites.google.com/view/programacion-unal/inicio>

Grupo de facebook <https://www.facebook.com/groups/maratonesunbogota>

Correos institucionales:

- Ivan Castellanos - iycastellanosm@unal.edu.co
- Diego Niquefa - dsniquefav@unal.edu.co
- Manuel Vergara - mavergarad@unal.edu.co
- Osman Jimenez - odjimenezg@unal.edu.co
- Jonathan Carrillo - joacarrilloco@unal.edu.co

Gracias

¿Preguntas?
