Clase 27/6/19

**Problema:**

Intercambio o “swap”:

A → Va A → Vb  
B → Vb B → Va

3°

2°

1°

Codificación:

void swap(int, int);

int main()

{

int x, y;

swap(x, y);

}

void swap(int &*a*, int &*b*)

{

std::cout << "a = ";

std::cin >> a;

std::cout << "b = ";

std::cin >> b;

int aux{a};

a = b;

b = aux;

std::cout << "------------------------\n";

std::cout << "a = " << a << '\n'

<< "b = " << b << '\n';

}

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Stack: |  | Swap |  |  |
|  | Main(x=2, y=3) | Main(x=3, y=2) | Main(x=3, y=2) |  |

La referencia a una variable no genera una nueva variable con el valor de la variable, por ejemplo, *x*, sino que es una “flecha” que apunta a la variable x, por lo que si decimos:

int main()

{

int x, y;

swap(x, y);

}

void swap(int &*a*, int &*b*)

{

a = b;

}

Cuando decimos **a = b** estamos en realidad diciendo que **x = y**

Podemos también hacerlo **como estructura**:

#include <iostream>

struct ParInt

{

int p, s;

};

void swap(ParInt &*x*);

int main()

{

ParInt par; *// Acá declaramos las variables x e y*

*// x sería par.p e y sería par.s*

std::cin >> par.p >> par.s;

swap(par);

std::cout << "------------------------\n";

std::cout << "Primero = " << par.p << '\n'

<< "Segundo = " << par.s << '\n';

}

void swap(ParInt &*x*) *// 'x' en este caso hace referencia a 'par' por lo que poner 'x.p' es lo mismo que 'par.p'*

{

int aux{x.p};

x.p = x.s;

x.s = aux;

}