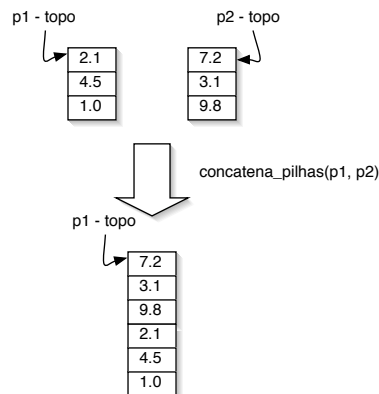


Considere a existência de um tipo abstrato Pilha de números de ponto flutuante, cuja interface está definida no arquivo `pilha.h` da seguinte forma:

```
typedef struct pilha Pilha;  
Pilha* cria(void);  
void push (Pilha* p, float v);  
float pop (Pilha* p);  
int vazia (Pilha* p);  
void libera (Pilha* p);
```

Sem conhecer a representação interna desse tipo abstrato `Pilha` e usando apenas as funções declaradas no arquivo `pilha.h`, implemente uma função que receba duas pilhas, `p1` e `p2`, e passe todos os elementos da pilha `p2` para o topo da pilha `p1`. A figura a seguir ilustra essa concatenação de pilhas:



Note que ao final dessa função, a pilha `p2` vai estar vazia e a pilha `p1` conterá todos os elementos das duas pilhas. Essa função deve obedecer o protótipo:

```
void concatena_pilhas (Pilha* p1, Pilha* p2);
```

Dica: Essa função pode ser implementada mais facilmente através de uma solução recursiva ou utilizando uma outra variável pilha auxiliar para fazer a transferência dos elementos entre as duas pilhas.