

Trabalho Prático

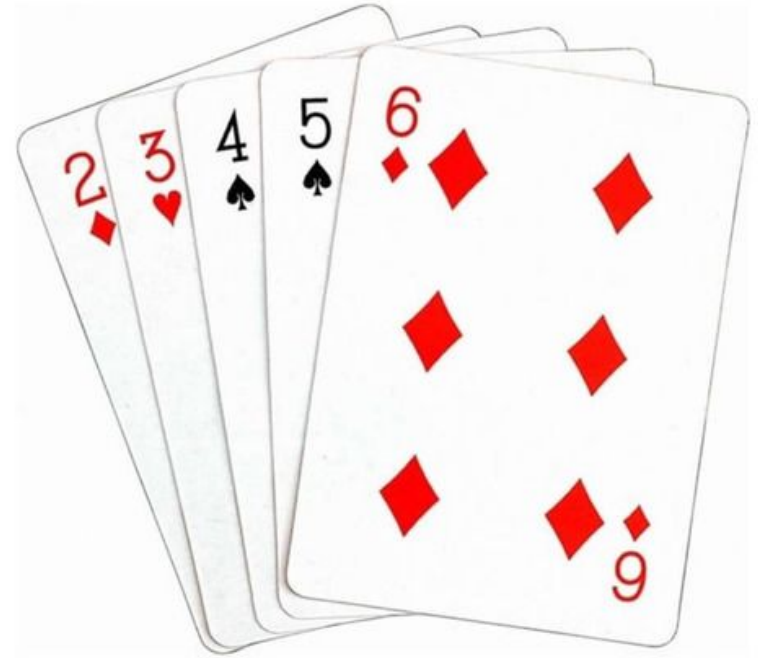
Estrutura de Dados 1:

Pife-pafe

Daniel Augusto Pires de Castro
Engenharia da Computação - Turma s71
Professor Rodrigo Minetto

Regras do jogo:

- Nove cartas na mão;
- Objetivo: formar três trincas;
 - Trinca: Conjunto de três cartas de mesmo número mas de naipes diferentes, ou de números consecutivos de mesmo naipe.
- A cada rodada, compra-se uma carta do cemitério ou do deck e descarta-se outra;
 - O topo do cemitério é visível.
- Formar duplas é interessante.



Motivação

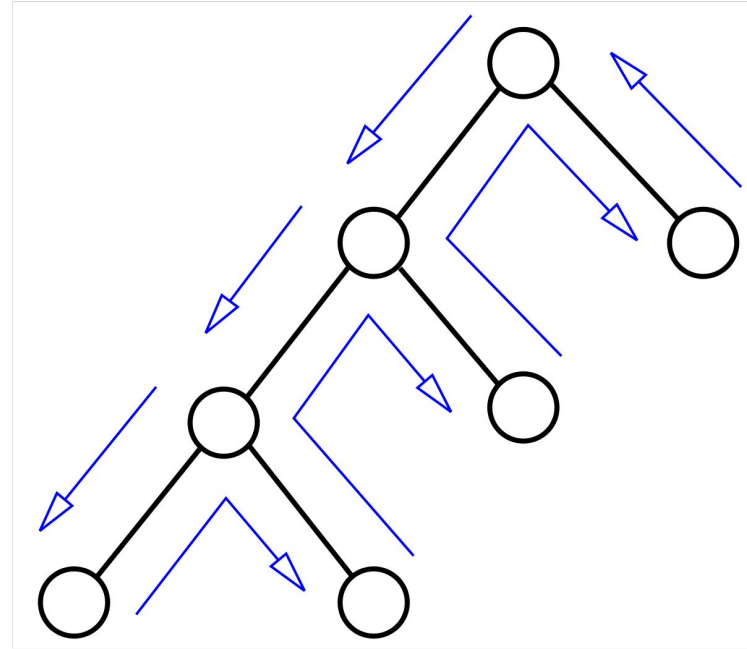
- A cada rodada, é possível realizar um número grande de combinações com as cartas da mão;
- Combinações favoráveis ao jogo podem ser determinadas através de um raciocínio matemático;
- Logo, há a possibilidade de gerar um algoritmo que “raciocine” tal como o cérebro.

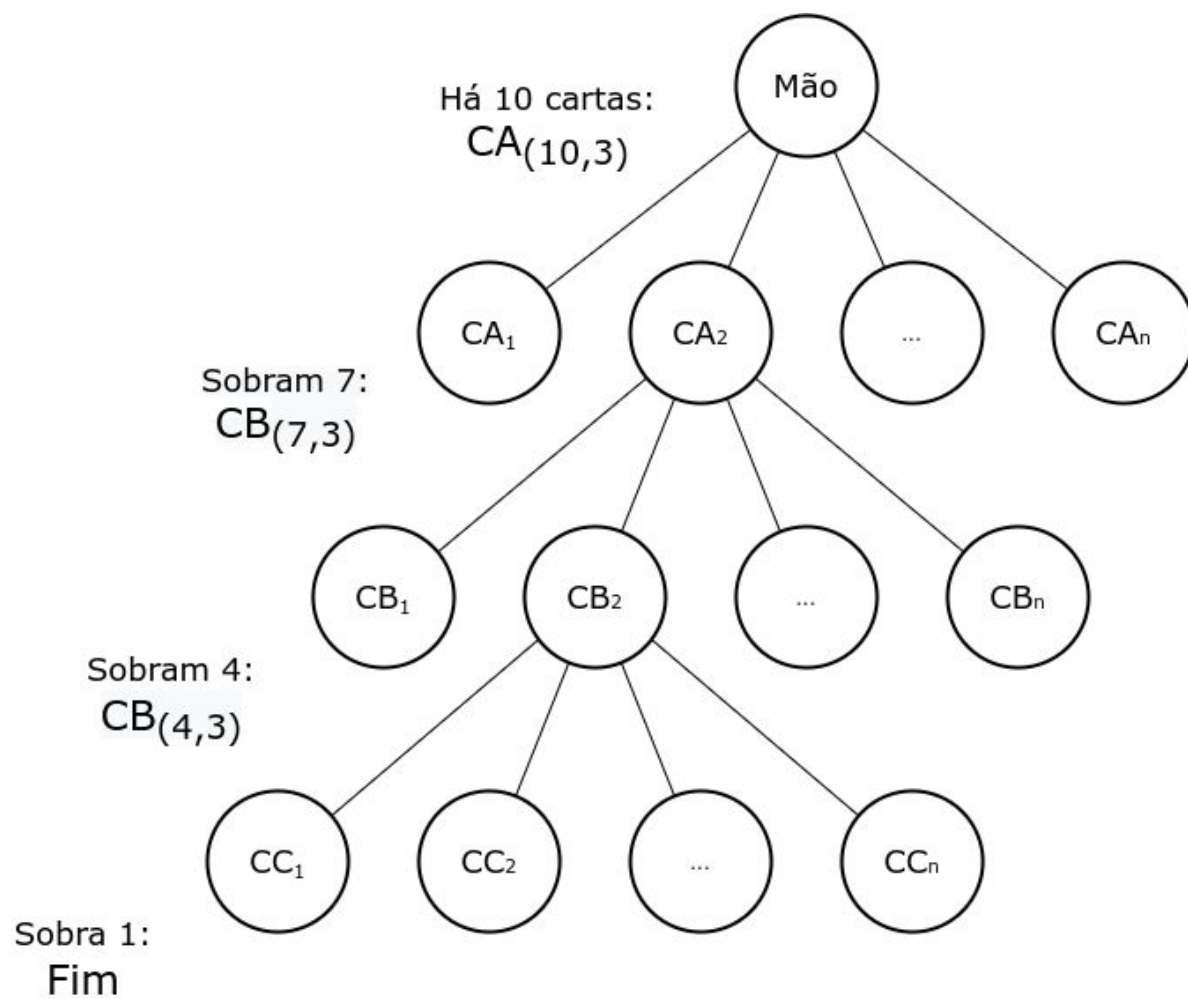
$$C_{n,p} = \frac{n!}{p!(n-p)!}$$

```
for ( i = 0; i < tam; i++)  
    for ( j = i + 1; j < tam; j++)  
        for ( k = j + 1; k < tam; k++)  
            // faz algo com i, j e k
```

Backtracking

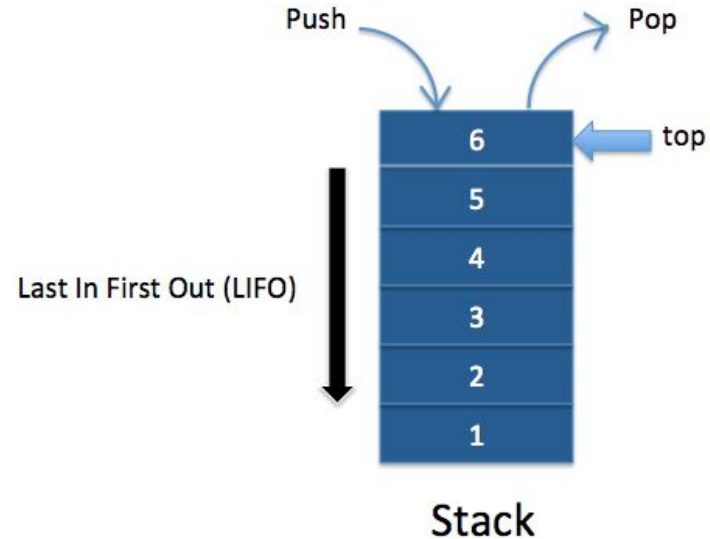
- “Backtracking é um tipo de algoritmo que representa um refinamento da busca por força bruta, em que múltiplas soluções podem ser eliminadas sem serem explicitamente examinadas.”;
 - Fonte: Wikipédia.
- Chamadas recursivas de função.



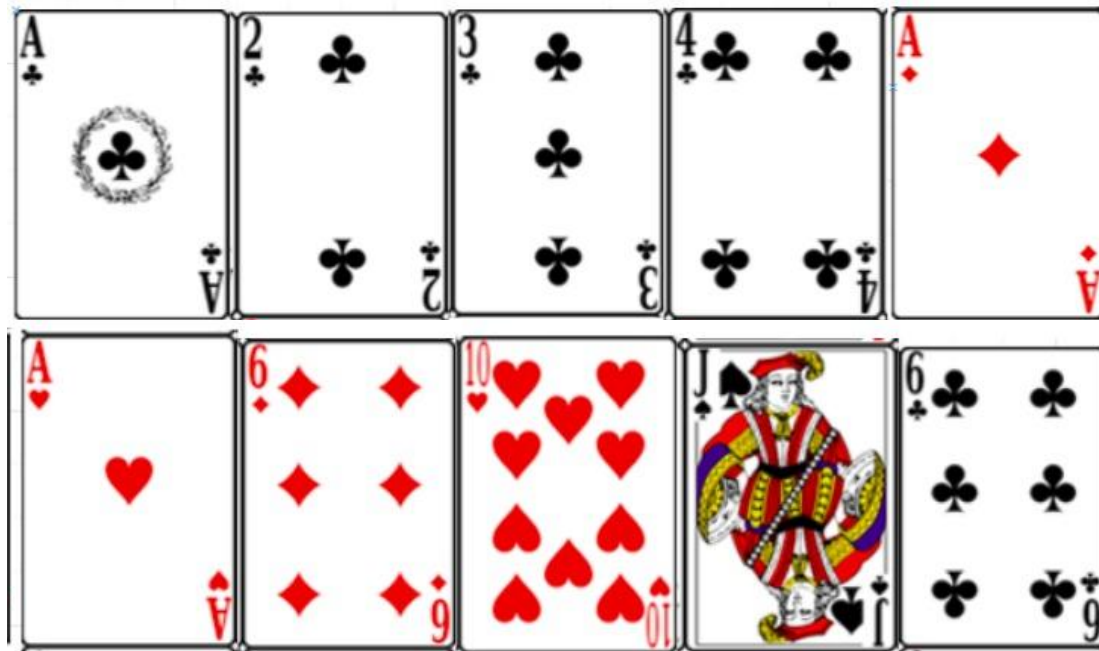


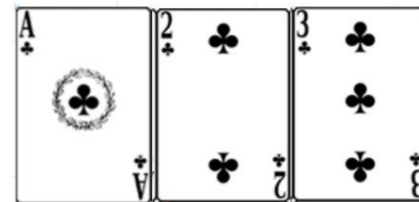
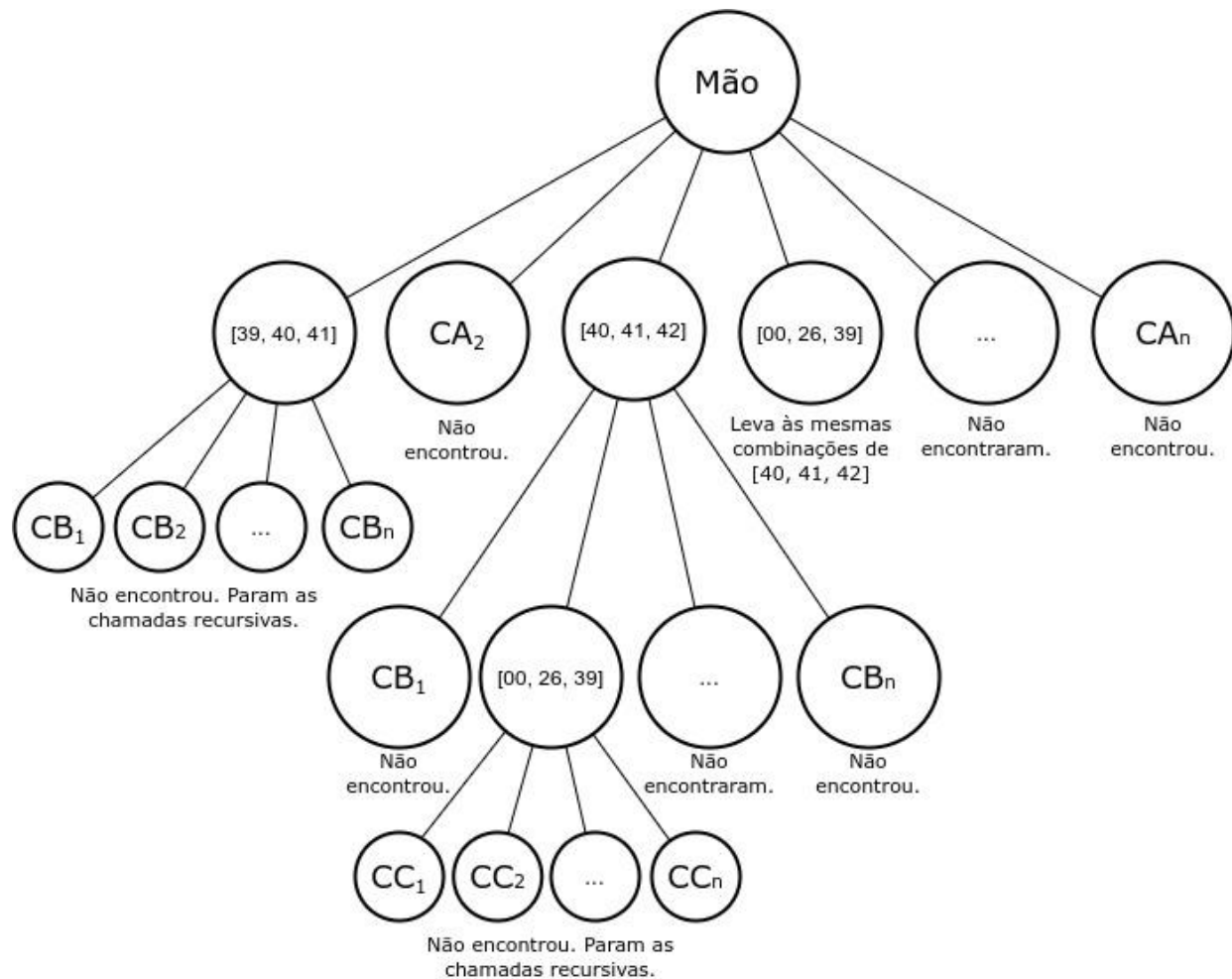
Pilhas

- Além do backtracking (que por si só já é uma pilha de chamadas recursivas), foi efetuado o uso extensivo de pilhas:
 - Cemitério do deck;
 - Pilhas de trincas e duplas;
 - Pilhas de combinações.
- Último elemento a ser inserido é o primeiro a ser retirado.

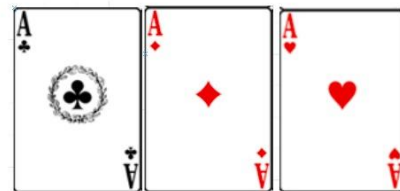


Exemplo hipotético de mão:

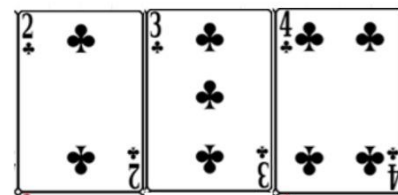





Pilha 1



Pilha 2





Implementação do algoritmo em Linguagem C

<https://github.com/daniapc/pife-pafe>

