Estruturas de Dados

Cristina Gomes Fernandes

Pilha

Lista linear em que todas as inserções e remoções são feitas numa mesma extremidade (topo).

Implementação sequencial:
um vetor $P[1\,..\,\text{MAX}]$ e uma variável inteira $topo$.

Operações:

- Inicialize$(P, topo)$
- Empilhe$(P, topo, x)$
- Desempilhe$(P, topo)$
- Topo$(P, topo)$
- Vazia$(P, topo)$

Implementação das operações

Inicialize $(P, topo)$

1 $topo \leftarrow 0$

Implementação das operações

Inicialize $(P, topo)$

1 $topo \leftarrow 0$

Empilhe $(P, topo, x)$

1 $topo \leftarrow topo + 1$

2 $P[topo] \leftarrow x$

Desempilhe $(P, topo)$

1 $topo \leftarrow topo - 1$

2 **devolva** $P[topo + 1]$

Implementação das operações

Inicialize $(P, topo)$

1 $topo \leftarrow 0$

Empilhe $(P, topo, x)$

1 $topo \leftarrow topo + 1$

2 $P[topo] \leftarrow x$

Desempilhe $(P, topo)$

1 $topo \leftarrow topo - 1$

2 **devolva** $P[topo + 1]$

Topo $(P, topo)$

1 **devolva** $P[topo]$

Implementação das operações

Inicialize $(P, topo)$

1 $topo \leftarrow 0$

Empilhe $(P, topo, x)$

1 $topo \leftarrow topo + 1$

2 $P[topo] \leftarrow x$

Desempilhe $(P, topo)$

1 $topo \leftarrow topo - 1$

2 **devolva** $P[topo + 1]$

Topo $(P, topo)$

1 **devolva** $P[topo]$

Vazia $(P, topo)$

1 **se** $topo = 0$

2 **então devolva** VERDADE

3 **senão devolva** FALSO

Aplicações

- cálculo de expressões em notação posfixa
- conversão de expressões em notação infixa para posfixa
- administração de memória em tempo de execução
- algoritmo de Graham para cálculo de casco convexo
- backtrack

Casco Convexo

Um conjunto S de pontos do plano é convexo se, para quaisquer dois pontos x e y de S, o segmento entre x e y está totalmente contido em S.

O casco convexo de um conjunto Q de pontos do plano é o menor conjunto convexo que contém todos os pontos de Q.

Problema: Dados n pontos do plano, determinar o casco convexo deles.

Algoritmo de Graham:

Entrada: um inteiro n e um vetor $p[1\,..\,n]$, com as coordenadas dos n pontos.
Saída: um vetor $q[1\,..\,t]$ com os t vértices do casco convexo.

Algoritmo de Graham

Primeira fase: rearranje $p[1\,..\,n]$ de modo que

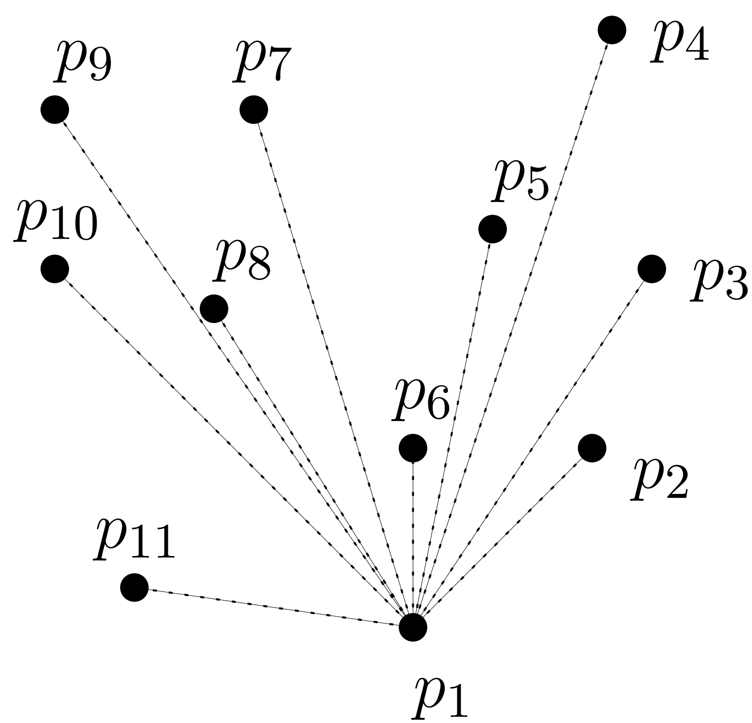
- $p[1]$ seja o ponto mais ao sul no plano;
- $p[2\,..\,n]$ estejam ordenados de acordo com o ângulo entre a reta horizontal passando por $p[1]$ e a reta passando por $p[1]$ e por $p[i]$, para $i = 2, \ldots, n$.

Algoritmo de Graham

Primeira fase: rearranje $p[1\,..\,n]$ de modo que

- $p[1]$ seja o ponto mais ao sul no plano;
- $p[2\,..\,n]$ estejam ordenados de acordo com o ângulo entre a reta horizontal passando por $p[1]$ e a reta passando por $p[1]$ e por $p[i]$, para $i = 2, \ldots, n$.

Exemplo:

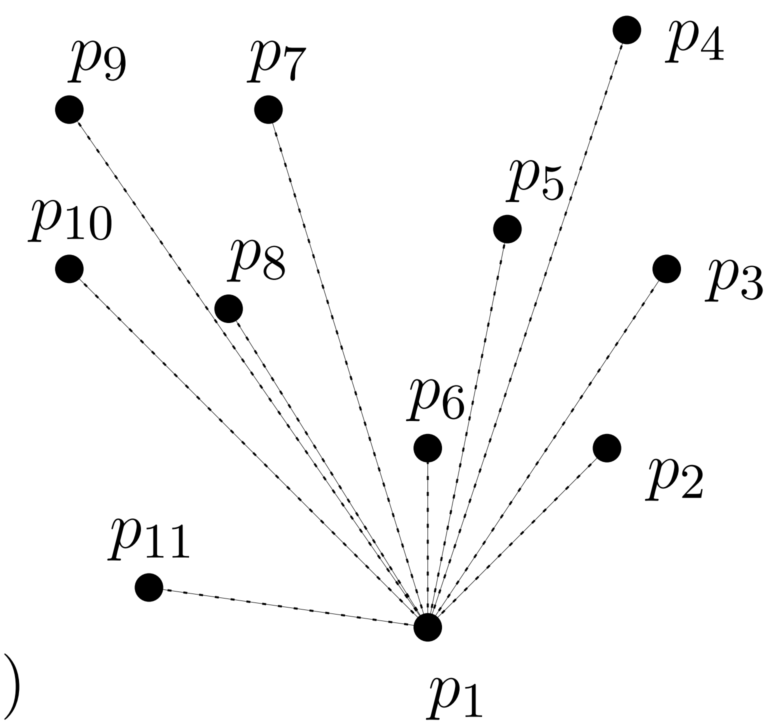


Algoritmo de Graham

Primeira fase: rearranje $p[1\,..\,n]$ de modo que

- $p[1]$ seja o ponto mais ao sul no plano;
- $p[2\,..\,n]$ estejam ordenados de acordo com o ângulo entre a reta horizontal passando por $p[1]$ e a reta passando por $p[1]$ e por $p[i]$, para $i = 2, \ldots, n$.

Exemplo:



Tempo consumido: $O(n \log n)$

Algoritmo de Graham

Casco Convexo (p, n)

Inicialize(St, t)
Empilhe$(St, t, 1)$
Empilhe$(St, t, 2)$
para $i \leftarrow 3$ **até** n **faça**
    enquanto Ângulo$(p, St[t-1], St[t], i) \geq 180^o$ **faça**
        Desempilhe(St, t)
    Empilhe(St, t, i)
devolva $St[1\,.\,.\,t]$

Ângulo (p, i, j, k) devolve o ângulo no sentido horário entre a semireta que começa em $p[j]$ e passa por $p[i]$ e a semireta que começa em $p[j]$ e passa por $p[k]$.

Algoritmo de Graham

Casco Convexo (p, n)

Inicialize(St, t)
Empilhe$(St, t, 1)$
Empilhe$(St, t, 2)$
para $i \leftarrow 3$ **até** n **faça**
    enquanto Ângulo$(p, St[t-1], St[t], i) \geq 180^o$ **faça**
        Desempilhe(St, t)
    Empilhe(St, t, i)
devolva $St[1\,..\,t]$

Quanto tempo consome o algoritmo de Graham em função de n?