

Universidade Federal de Alfenas

Projeto e Análise de Algoritmos

Aula 04 – Introdução a Análise de Algoritmos

humberto@bcc.unifal-mg.edu.br



Última aula

- Fundamentos de Matemática
- Exercícios:
 - Somatórios;
 - Logaritmos e Expoentes;
 - Limites.

Análise de Algoritmos

- O que é analisar um algoritmo?

Análise de Algoritmos

- O que é analisar um algoritmo?
 - Analisar significa prever os recursos que um algoritmo necessitará.

Análise de Algoritmos

- O que é analisar um algoritmo?
 - Analisar significa prever os recursos que um algoritmo necessitará.
- O que a análise nos fornece?

Análise de Algoritmos

- O que é analisar um algoritmo?
 - Analisar significa prever os recursos que um algoritmo necessitará.
- O que a análise nos fornece?
 - Condições de:
 - Determinar se o algoritmo é viável para a sua aplicação;
 - Comparar diversos algoritmos para o mesmo problema.

Análise de Algoritmos

- O que é analisar um algoritmo?
 - Analisar significa prever os recursos que um algoritmo necessitará.
- O que a análise nos fornece?
 - Condições de:
 - Determinar se o algoritmo é viável para a sua aplicação;
 - Comparar diversos algoritmos para o mesmo problema.
- Quais são as técnicas de análises?

Análise de Algoritmos

- O que é analisar um algoritmo?
 - Analisar significa prever os recursos que um algoritmo necessitará.
- O que a análise nos fornece?
 - Condições de:
 - Determinar se o algoritmo é viável para a aplicação;
 - Comparar diversos algoritmos para o mesmo problema.
- Quais são as técnicas de análises?
 - Experimentação;
 - Análise assintótica;
 - Dentre outras...

Análise de Algoritmos

- O **ferramental básico** de análise que utilizaremos ao longo da disciplina **envolve** a caracterização do:



Análise de Algoritmos

- O **ferramental básico** de análise que utilizaremos ao longo da disciplina **envolve** a caracterização do:
 - **tempo** de execução de algoritmos;



Análise de Algoritmos

- O **ferramental básico** de análise que utilizaremos ao longo da disciplina **envolve** a caracterização do:
 - **tempo** de execução de algoritmos;
 - Consumo de **memória**.



Análise de Algoritmos

- O **ferramental básico** de análise que utilizaremos ao longo da disciplina **envolve** a caracterização do:
 - **tempo** de execução de algoritmos;
 - Consumo de **memória**.
- O **tempo** de execução é uma **medida básica**, pois tempo é um recurso precioso.
 - Os usuários finais muitas vezes medem a qualidade dos programas comerciais em função do tempo de execução dos mesmos.



Análise de Algoritmos

- O **ferramental básico** de análise que utilizaremos ao longo da disciplina **envolve** a caracterização do:
 - **tempo** de execução de algoritmos;
 - Consumo de **memória**.
- O **tempo** de execução é uma **medida básica**, pois tempo é um recurso precioso.
 - Os usuários finais muitas vezes medem a qualidade dos programas comerciais em função do tempo de execução dos mesmos.
- Atualmente, **poucos sistemas se preocupam com o consumo de memória**, se compararmos com o passado recente (década de 80).

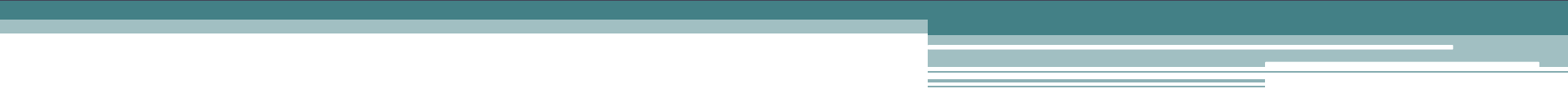


Análise de Algoritmos

- O **ferramental básico** de análise que utilizaremos ao longo da disciplina **envolve** a caracterização do:
 - **tempo** de execução de algoritmos;
 - Consumo de **memória**.
- O **tempo** de execução é uma **medida básica**, pois tempo é um recurso precioso.
 - Os usuários finais muitas vezes medem a qualidade dos programas comerciais em função do tempo de execução dos mesmos.
- Atualmente, **poucos sistemas se preocupam com o consumo de memória**, se compararmos com o passado recente (década de 80).
- Mas...
 - Muitos sistemas atuais precisam de muita memória.



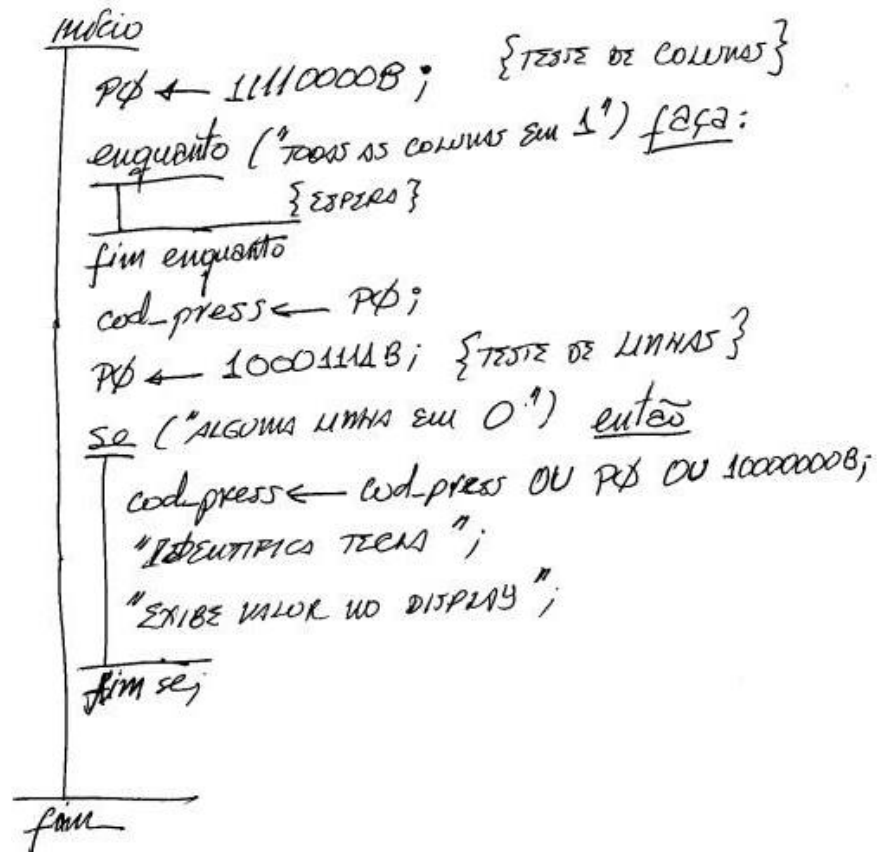
Metodologias para análise de algoritmos

A series of horizontal lines in teal and light blue colors, with varying lengths and offsets, creating a modern, layered effect across the middle of the slide.

Metodologias para análise de algoritmos

- O tempo de execução de um algoritmo ou uma operação sobre uma estrutura de dados depende de uma série de fatores...

- Citem tais fatores...



Metodologias para análise de algoritmos

- O tempo depende, por exemplo do(a):
 - Considerando o Hardware:

Metodologias para análise de algoritmos

- O tempo depende, por exemplo do(a):
 - Considerando o Hardware:
 - **Processador** utilizado;

Metodologias para análise de algoritmos

- O tempo depende, por exemplo do(a):
 - Considerando o Hardware:
 - Processador utilizado;
 - Da quantidade de processadores disponíveis;

Metodologias para análise de algoritmos

- O tempo depende, por exemplo do(a):
 - Considerando o Hardware:
 - Processador utilizado;
 - Da quantidade de processadores disponíveis;
 - **Frequência do processador;**

Metodologias para análise de algoritmos

- O tempo depende, por exemplo do(a):
 - Considerando o Hardware:
 - Processador utilizado;
 - Da quantidade de processadores disponíveis;
 - Frequência do processador;
 - Ciclos por instrução;

Metodologias para análise de algoritmos

- O tempo depende, por exemplo do(a):
 - Considerando o Hardware:
 - Processador utilizado;
 - Da quantidade de processadores disponíveis;
 - Frequência do processador;
 - Ciclos por instrução;
 - Disco;

Metodologias para análise de algoritmos

- O tempo depende, por exemplo do(a):
 - Considerando o Hardware:
 - Processador utilizado;
 - Da quantidade de processadores disponíveis;
 - Frequência do processador;
 - Ciclos por instrução;
 - Disco;
 - Memória (Primária, Secundária);

Metodologias para análise de algoritmos

- O tempo depende, por exemplo do(a):
 - Considerando o Hardware:
 - Processador utilizado;
 - Da quantidade de processadores disponíveis;
 - Frequência do processador;
 - Ciclos por instrução;
 - Disco;
 - Memória (Primária, Secundária);
 - Dentre outros fatores!!!

Metodologias para análise de algoritmos

- Dependência a nível de Software:
 - Sistema operacional
 - Qual é o kernel do S.O.;

Metodologias para análise de algoritmos

- Dependência a nível de Software:
 - Sistema operacional
 - Qual é o kernel do S.O.;
 - Quantidade de processos na fila de espera do algoritmo de escalonamento;

Metodologias para análise de algoritmos

- Dependência a nível de Software:
 - Sistema operacional
 - Qual é o kernel do S.O.;
 - Quantidade de processos na fila de espera do algoritmo de escalonamento;
 - Do próprio algoritmo de escalonamento;

Metodologias para análise de algoritmos

- Dependência a nível de Software:
 - Sistema operacional
 - Qual é o kernel do S.O.;
 - Quantidade de processos na fila de espera do algoritmo de escalonamento;
 - Do próprio algoritmo de escalonamento;
 - Da prioridade de cada processo;

Metodologias para análise de algoritmos

- Dependência a nível de Software:
 - Sistema operacional
 - Qual é o kernel do S.O.;
 - Quantidade de processos na fila de espera do algoritmo de escalonamento;
 - Do próprio algoritmo de escalonamento;
 - Da prioridade de cada processo;
 - O processo (ou parte dele) está em memória primária ou secundária???

Metodologias para análise de algoritmos

- Dependência a nível de Software:
 - Sistema operacional
 - Qual é o kernel do S.O.;
 - Quantidade de processos na fila de espera do algoritmo de escalonamento;
 - Do próprio algoritmo de escalonamento;
 - Da prioridade de cada processo;
 - O processo (ou parte dele) está em memória primária ou secundária???
 - Qual é o quantum dado ao processo?

Metodologias para análise de algoritmos

- Dependência a nível de Software:
 - Sistema operacional
 - Qual é o kernel do S.O.;
 - Quantidade de processos na fila de espera do algoritmo de escalonamento;
 - Do próprio algoritmo de escalonamento;
 - Da prioridade de cada processo;
 - O processo (ou parte dele) está em memória primária ou secundária???
 - Qual é o quantum dado ao processo?
 - Troca de contexto.

Metodologias para análise de algoritmos

- Dependência a nível de Software:
 - Sistema operacional
 - Qual é o kernel do S.O.;
 - Quantidade de processos na fila de espera do algoritmo de escalonamento;
 - Do próprio algoritmo de escalonamento;
 - Da prioridade de cada processo;
 - O processo (ou parte dele) está em memória primária ou secundária???
 - Qual é o quantum dado ao processo?
 - Troca de contexto.
 - Da linguagem utilizada pelo programador;

Metodologias para análise de algoritmos

- Dependência a nível de Software:
 - Sistema operacional
 - Qual é o kernel do S.O.;
 - Quantidade de processos na fila de espera do algoritmo de escalonamento;
 - Do próprio algoritmo de escalonamento;
 - Da prioridade de cada processo;
 - O processo (ou parte dele) está em memória primária ou secundária???
 - Qual é o quantum dado ao processo?
 - Troca de contexto.
 - Da linguagem utilizada pelo programador;
 - Do compilador utilizado pelo programador:
 - Com otimização, sem otimização...

Metodologias para análise de algoritmos

- Dependência a nível de Software:
 - Sistema operacional
 - Qual é o kernel do S.O.;
 - Quantidade de processos na fila de espera do algoritmo de escalonamento;
 - Do próprio algoritmo de escalonamento;
 - Da prioridade de cada processo;
 - O processo (ou parte dele) está em memória primária ou secundária???
 - Qual é o quantum dado ao processo?
 - Troca de contexto.
 - Da linguagem utilizada pelo programador;
 - Do compilador utilizado pelo programador:
 - Com otimização, sem otimização...
 - Da habilidade do programador em utilizar chamadas mais rápidas (ou não);

Metodologias para análise de algoritmos

- Dependência a nível de Software:

- Sistema operacional
 - Qual é o kernel do S.O.;
 - Quantidade de processos na fila de espera do algoritmo de escalonamento;
 - Do próprio algoritmo de escalonamento;
 - Da prioridade de cada processo;
 - O processo (ou parte dele) está em memória primária ou secundária???
 - Qual é o quantum dado ao processo?
 - Troca de contexto.
- Da linguagem utilizada pelo programador;
- Do compilador utilizado pelo programador:
 - Com otimização, sem otimização...
- Da habilidade do programador em utilizar chamadas mais rápidas (ou não);
- Da máquina virtual que executa o programa;

Metodologias para análise de algoritmos

- Dependência a nível de Software:

- Sistema operacional
 - Qual é o kernel do S.O.;
 - Quantidade de processos na fila de espera do algoritmo de escalonamento;
 - Do próprio algoritmo de escalonamento;
 - Da prioridade de cada processo;
 - O processo (ou parte dele) está em memória primária ou secundária???
 - Qual é o quantum dado ao processo?
 - Troca de contexto.
- Da linguagem utilizada pelo programador;
- Do compilador utilizado pelo programador:
 - Com otimização, sem otimização...
- Da habilidade do programador em utilizar chamadas mais rápidas (ou não);
- Da máquina virtual que executa o programa;
- Etc... Etc... Etc... Etc... Etc... Etc... Etc... Etc... Etc...

Metodologias para análise de algoritmos

- Dependência considerando a entrada de dados:
 - Depende da natureza do problema....

Metodologias para análise de algoritmos

- Considerando esta grande variedade de fatores que influenciam o tempo de um algoritmo, qual seria a maneira mais adequada de medi-lo?

Metodologias para análise de algoritmos

- Considerando esta grande variedade de fatores que influenciam o tempo de um algoritmo, qual seria a maneira mais adequada de medi-lo?
- Se tal algoritmo foi implementado, podemos estudar o tempo gasto por ele executando-o com vários dados de entrada e registrando o tempo gasto por cada execução;

Metodologias para análise de algoritmos

- Considerando esta grande variedade de fatores que influenciam o tempo de um algoritmo, qual seria a maneira mais adequada de medi-lo?
- Se tal algoritmo foi implementado, podemos estudar o tempo gasto por ele executando-o com vários dados de entrada e registrando o tempo gasto por cada execução;
- Em geral, estamos interessados em determinar a dependência do tempo de execução com respeito ao tamanho/distribuição da entrada fornecida ao algoritmo.

Metodologias para análise de algoritmos

- Considerando esta grande variedade de fatores que influenciam o tempo de um algoritmo, qual seria a maneira mais adequada de medi-lo?
- Se tal algoritmo foi implementado, podemos estudar o tempo gasto por ele executando-o com vários dados de entrada e registrando o tempo gasto por cada execução;
- Em geral, estamos interessados em determinar a dependência do tempo de execução com respeito ao tamanho/distribuição da entrada fornecida ao algoritmo.
- Citem exemplos de entradas de algoritmos...

Metodologias para análise de algoritmos

- Considerando o algoritmo **escalonador de processos**:
 - A entrada é o **conjunto de processos prontos para execução** no S.O., por exemplo...

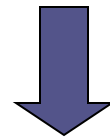
```
top - 19:18:23 up 27 days, 10:47, 1 user, load average: 0.00, 0.00, 0.00
Tasks: 115 total, 1 running, 112 sleeping, 1 stopped, 1 zombie
Cpu(s): 0.2%us, 0.0%sy, 0.0%ni, 99.8%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 1025648k total, 509988k used, 515660k free, 114612k buffers
Swap: 1951856k total, 40252k used, 1911604k free, 185928k cached
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
15117	humberto	20	0	2308	1128	852	R	1	0.1	0:13.32	top
1	root	20	0	2844	1688	544	S	0	0.2	0:01.14	init
2	root	15	-5	0	0	0	S	0	0.0	0:00.00	kthreadd
3	root	RT	-5	0	0	0	S	0	0.0	0:00.04	migration/0
4	root	15	-5	0	0	0	S	0	0.0	0:00.14	ksoftirqd/0
5	root	RT	-5	0	0	0	S	0	0.0	0:00.00	watchdog/0
6	root	RT	-5	0	0	0	S	0	0.0	0:00.04	migration/1
7	root	15	-5	0	0	0	S	0	0.0	0:00.08	ksoftirqd/1
8	root	RT	-5	0	0	0	S	0	0.0	0:00.00	watchdog/1
9	root	15	-5	0	0	0	S	0	0.0	0:04.86	events/0
10	root	15	-5	0	0	0	S	0	0.0	0:04.88	events/1
11	root	15	-5	0	0	0	S	0	0.0	0:00.00	khelper
46	root	15	-5	0	0	0	S	0	0.0	0:01.98	kblockd/0
47	root	15	-5	0	0	0	S	0	0.0	0:00.50	kblockd/1
50	root	15	-5	0	0	0	S	0	0.0	0:00.00	kacpid
51	root	15	-5	0	0	0	S	0	0.0	0:00.00	kacpi_notify
130	root	15	-5	0	0	0	S	0	0.0	0:00.00	kseriod
170	root	20	0	0	0	0	S	0	0.0	0:00.58	pdflush
171	root	20	0	0	0	0	S	0	0.0	0:09.00	pdflush
172	root	15	-5	0	0	0	S	0	0.0	0:01.68	kswapd0
213	root	15	-5	0	0	0	S	0	0.0	0:00.00	aio/0
214	root	15	-5	0	0	0	S	0	0.0	0:00.00	aio/1
878	www-data	20	0	26392	13m	4512	S	0	1.3	0:07.36	apache2
923	www-data	20	0	30512	17m	4412	S	0	1.7	0:05.30	apache2

Metodologias para análise de algoritmos

- Considerando um algoritmo de ordenação...
 - A entrada é o conjunto de elementos a serem ordenados...

5 1 12 -5 16 2 12 14



-5 1 2 5 12 12 14 16

Metodologias para análise de algoritmos

- Considerando o tamanho da entrada, então podemos medir o tempo gasto pelo algoritmo executando-o diversas vezes, com diversas entradas...

Entrada 1:

5

Entrada 2:

12

Entrada 3:

16	2
----	---

Entrada 4:

2	12	14
---	----	----

Entrada 5:

2	12	14
---	----	----

Entrada 6:

12	-5	16	2
----	----	----	---

Entrada 7:

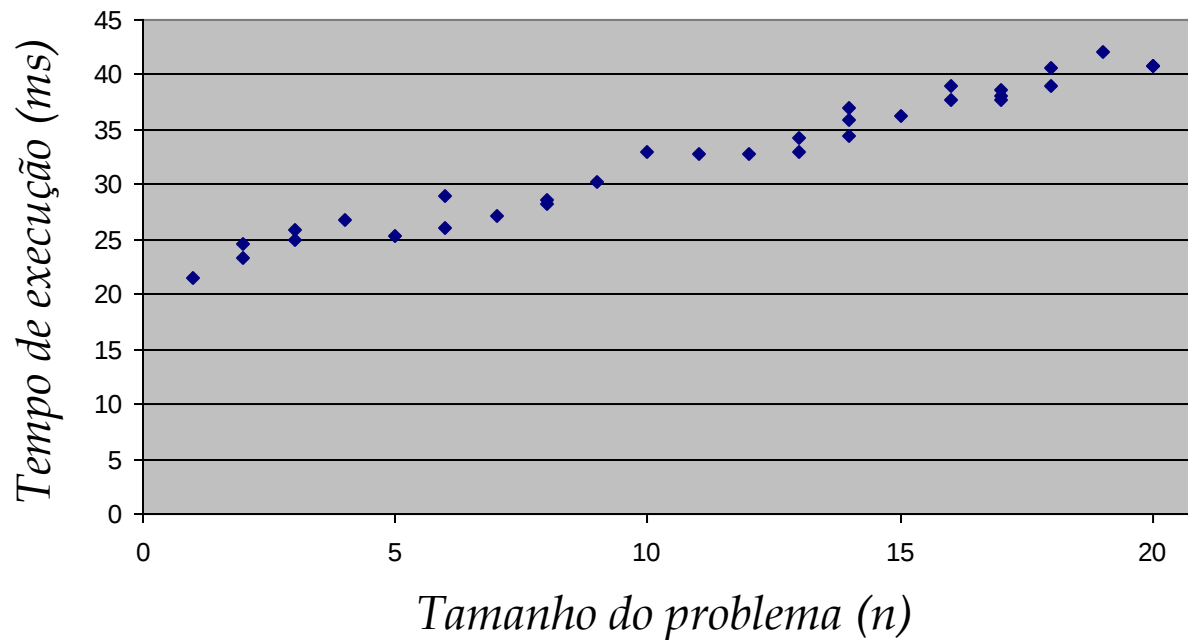
-5	16	2	12	14
----	----	---	----	----

Etc...

12	-5	16	2	12	14
----	----	----	---	----	----

Metodologias para análise de algoritmos

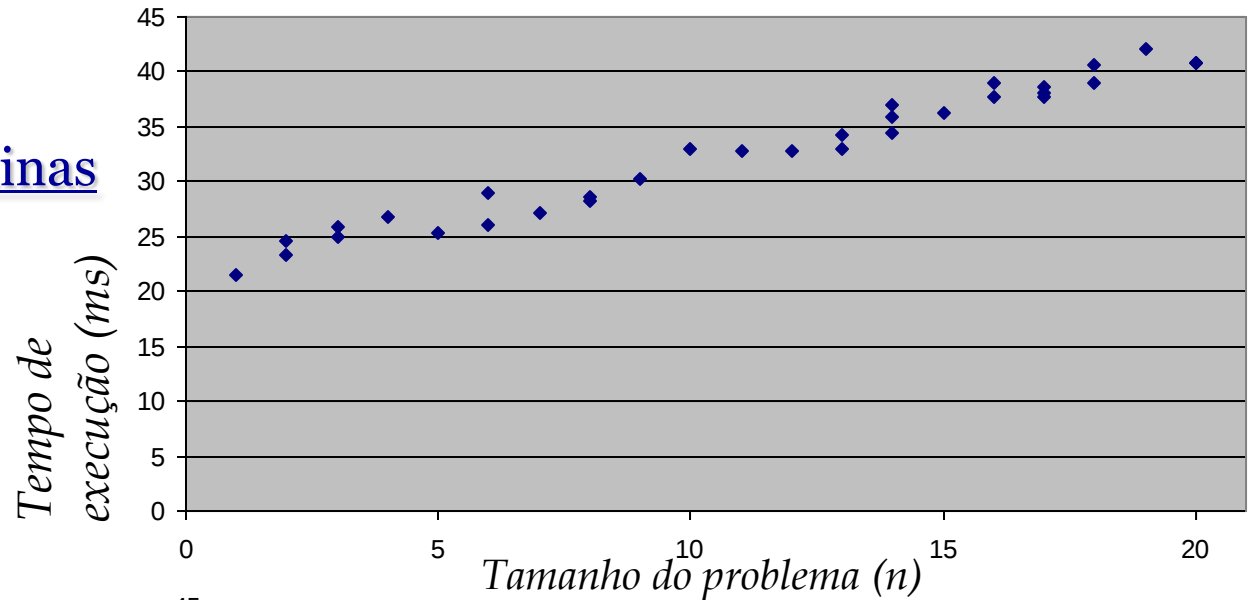
- Podemos então visualizar os resultados desse experimento plotando o desempenho de cada execução do algoritmo como um ponto com coordenada x igual ao tamanho da entrada e coordenada y igual ao tempo de execução, ou igual a quantidade de memória utilizada.



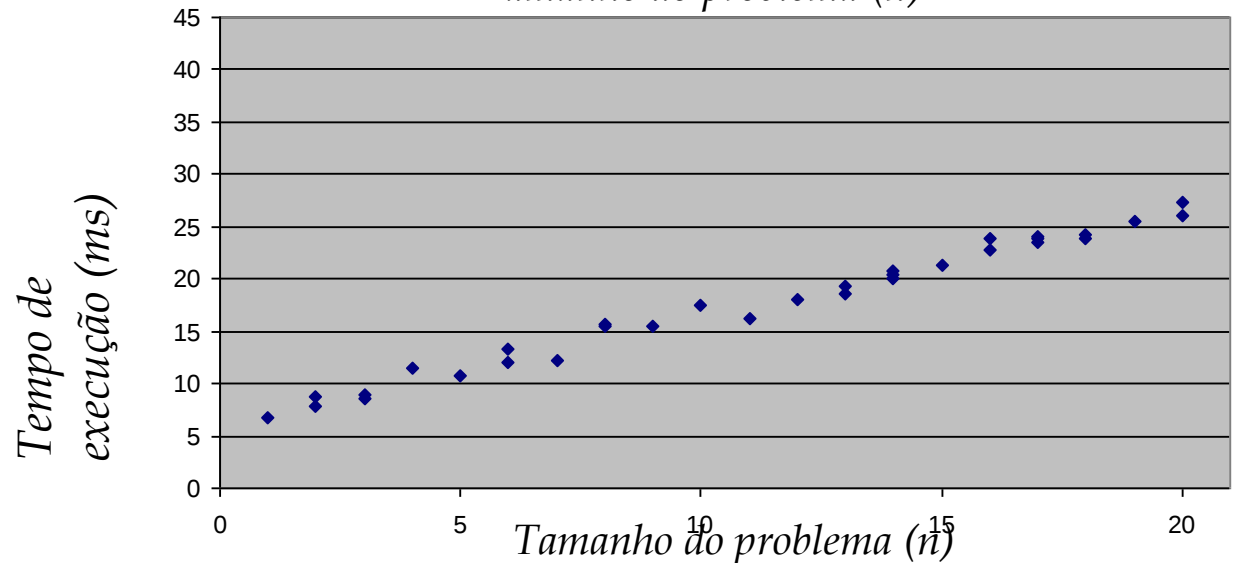
Metodologias para análise de algoritmos

- Exemplo do mesmo experimento em máquinas distintas....

- Pentium 4
3.0 GHz
2 GB RAM



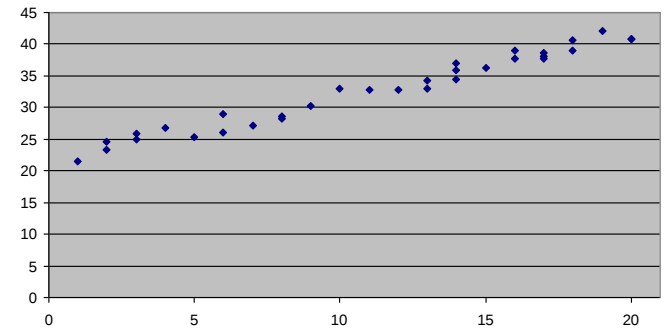
- Core2Quad 2.0 GHz
3 GB RAM



Metodologias para análise de algoritmos

Análise por tempo de execução:

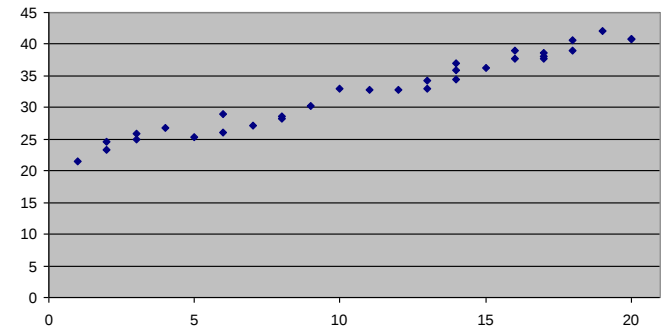
- Esta análise exige:



Metodologias para análise de algoritmos

Análise por tempo de execução:

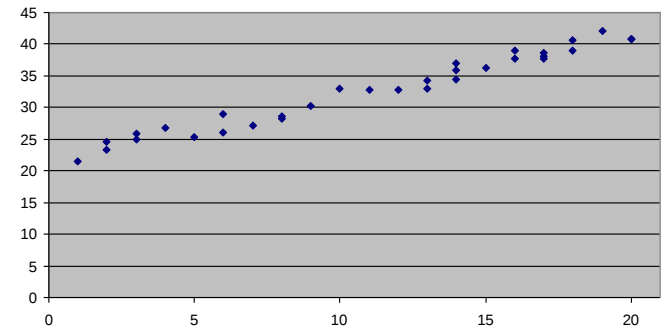
- Esta análise exige:
 - a escolha de um bom conjunto de entradas;



Metodologias para análise de algoritmos

Análise por tempo de execução:

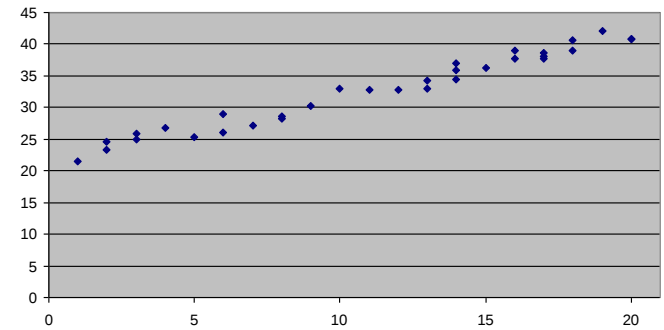
- Esta análise exige:
 - a escolha de um bom conjunto de entradas;
 - Que um número de testes suficientes devem ser executados para que possam ser feitas afirmações válidas sob um ponto de vista estatístico;
 - Pois **muita coisa** vai influenciar em cada execução...
 - Por exemplo...



Metodologias para análise de algoritmos

Análise por tempo de execução:

- Esta análise exige:
 - a escolha de um bom conjunto de entradas;
 - Que um número de testes suficientes devem ser executados para que possam ser feitas afirmações válidas sob um ponto de vista estatístico;
 - Pois muita coisa vai influenciar em cada execução...
 - Por exemplo...
- Ambos pontos são sempre questionáveis, de acordo com o ponto de vista do analista.



Metodologias para análise de algoritmos

- Estudos experimentais dos tempos de execução são úteis, mas eles possuem algumas limitações e/ou dificuldades:

Metodologias para análise de algoritmos

- Estudos experimentais dos tempos de execução são úteis, mas eles possuem algumas limitações e/ou dificuldades:
 - Experimentos podem ser feitos apenas em um **número limitado de entradas** de teste **para a grande maioria dos problemas**;
 - Geralmente a quantidade de entradas possíveis é exponencial em função do tamanho da entrada do problema;

Metodologias para análise de algoritmos

- Estudos experimentais dos tempos de execução são úteis, **mas** eles possuem algumas limitações e/ou dificuldades:
 - Experimentos podem ser feitos apenas em um **número limitado de entradas** de teste **para a grande maioria dos problemas**;
 - Geralmente a quantidade de entradas possíveis é exponencial em função do tamanho da entrada do problema;
 - **O conjunto de teste deve ser representativo**; (como garantir???)

Metodologias para análise de algoritmos

- Estudos experimentais dos tempos de execução são úteis, **mas** eles possuem algumas limitações e/ou dificuldades:
 - Experimentos podem ser feitos apenas em um **número limitado de entradas** de teste **para a grande maioria dos problemas**;
 - Geralmente a quantidade de entradas possíveis é exponencial em função do tamanho da entrada do problema;
 - O conjunto de teste deve ser **representativo**; (como garantir???)
 - É difícil comparar a eficiência de dois algoritmos a não ser que os experimentos para a obtenção de seus tempos de execução tenham sido feitos com o **mesmo hardware e software**;
//além disso o S.O. é dinâmico...

Metodologias para análise de algoritmos

- Estudos experimentais dos tempos de execução são úteis, **mas** eles possuem algumas limitações e/ou dificuldades:
 - Experimentos podem ser feitos apenas em um **número limitado de entradas** de teste **para a grande maioria dos problemas**;
 - Geralmente a quantidade de entradas possíveis é exponencial em função do tamanho da entrada do problema;
 - O conjunto de teste deve ser representativo; (como garantir???)
 - É difícil comparar a eficiência de dois algoritmos a não ser que os experimentos para a obtenção de seus tempos de execução tenham sido feitos com o **mesmo hardware e software**;
//além disso o S.O. é dinâmico...
 - **É necessário implementar e executar** um algoritmo para poder estudar seu tempo de execução experimentalmente.

Metodologias para análise de algoritmos

- Assim, a **experimentação possui um papel importante** na análise de algoritmos, **mas ela sozinha não é suficiente**.
- Portanto, desejamos uma metodologia analítica que:

Metodologias para análise de algoritmos

- Assim, a experimentação possui um papel importante na análise de algoritmos, mas ela sozinha não é suficiente.
- Portanto, desejamos uma metodologia analítica que:
 - Leve em conta **TODAS as entradas possíveis;**

Metodologias para análise de algoritmos

- Assim, a **experimentação possui um papel importante** na análise de algoritmos, **mas ela sozinha não é suficiente**.
- Portanto, desejamos uma metodologia analítica que:
 - Leve em conta **TODAS** as entradas possíveis;
 - **Permita-nos avaliar** a eficiência relativa de dois algoritmos de **uma forma independente do *hardware* e do *software*** onde eles são executados;

Metodologias para análise de algoritmos

- Assim, a experimentação possui um papel importante na análise de algoritmos, mas ela sozinha não é suficiente.
- Portanto, desejamos uma metodologia analítica que:
 - Leve em conta **TODAS** as entradas possíveis;
 - **Permita-nos avaliar** a eficiência relativa de dois algoritmos **de** uma forma independente do *hardware* e do *software* onde eles são executados;
 - **Possa ser feita** através do estudo de uma descrição em alto nível do algoritmo **sem que seja necessário implementá-lo ou executá-lo.**

Metodologias para análise de algoritmos

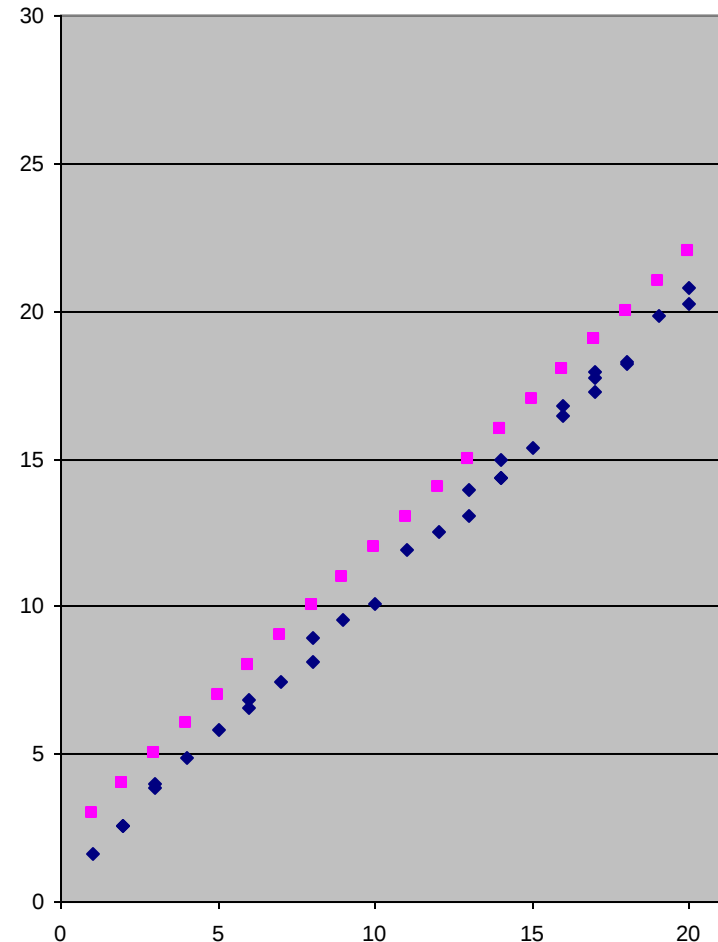
- Esta metodologia simplificada associa uma função $f(n)$ a cada algoritmo para descrever o crescimento do tempo de execução f em função do tamanho da entrada n .

Metodologias para análise de algoritmos

- Esta metodologia simplificada associa uma função $f(n)$ a cada algoritmo para descrever o crescimento do tempo de execução f em função do tamanho da entrada n .
- Exemplo: A função pode ser um polinômio de primeiro grau:
 - $f(n) = c_1 \cdot n + c_2$
 - O que pode determinar na prática os valores de c_1 e c_2 ?

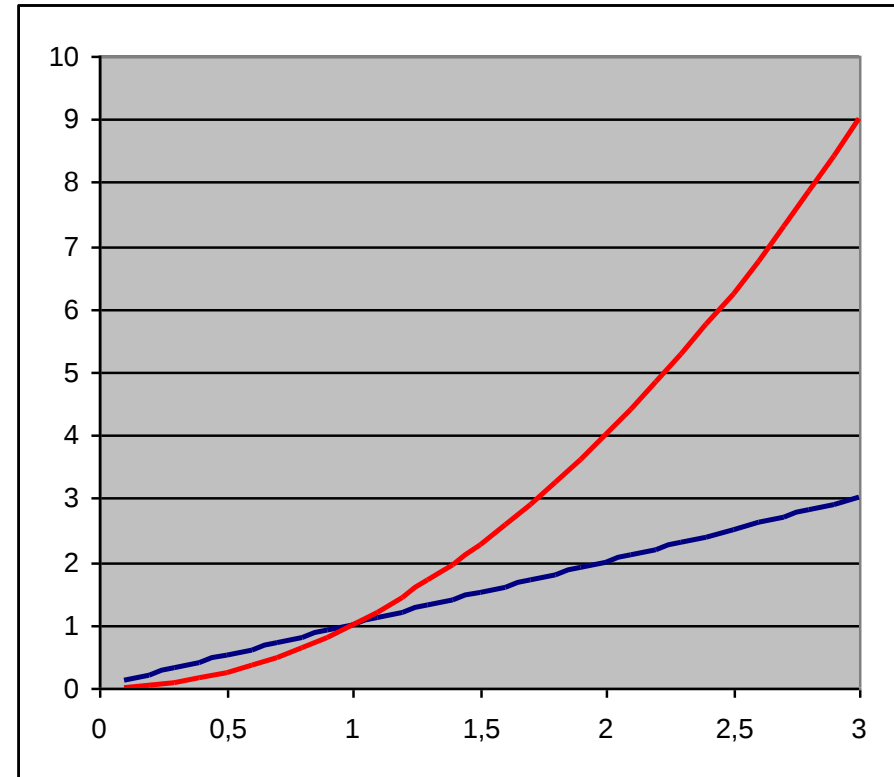
Metodologias para análise de algoritmos

- Funções típicas que serão encontradas incluem n e n^2 .
- Por exemplo:
 - O algoritmo A é executado em tempo proporcional a n .
 - Isso quer dizer que se fossemos realizar experimentos, constataríamos que o tempo de execução do algoritmo A para qualquer entrada de tamanho n nunca excede cn , onde **c é uma constante que depende do hardware e do software** usados no experimento.



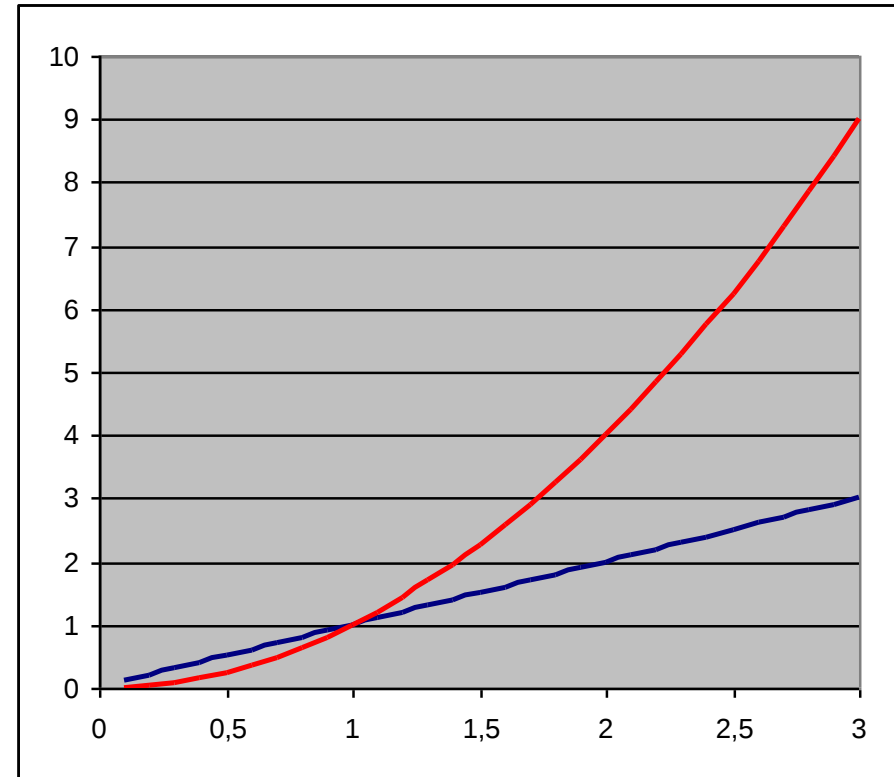
Metodologias para análise de algoritmos

- Dados dois algoritmos:
 - A: o tempo de execução cresce em função de n ;
 - B: o tempo de execução cresce em função de n^2 .



Metodologias para análise de algoritmos

- Dados dois algoritmos:
 - A: o tempo de execução cresce em função de n ;
 - B: o tempo de execução cresce em função de n^2 .
- Preferimos o algoritmo A ao algoritmo B, uma vez que a função n cresce mais lentamente que a função n^2 .



Próxima aula

- Uma **linguagem** para descrição de algoritmos;
 - Um **modelo computacional** para execução de algoritmos;
 - Uma **métrica** para medir o tempo de execução de algoritmos;
 - **Notação assintótica**.
- Leitura: Livro do Nivio Ziviani: Projeto de Algoritmos
 - 1 Introdução
 - 1.1 Algoritmos, Estruturas de Dados e Programas;
 - 1.2 Tipos de Dados e Tipos Abstrato de Dados;
 - 1.3 Medida de Tempo de Execução de um Programa;
 - 1.3.1 Comportamento Assintótico de Funções;
 - 1.3.2 Classes de Comportamento Assintótico
 - 1.4 Técnicas de Análise de Algoritmos

Bibliografia

- CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; (2002). Algoritmos – Teoria e Prática. Tradução da 2ª edição americana. Rio de Janeiro. Editora Campus.
- TAMASSIA, ROBERTO; GOODRICH, MICHAEL T. (2004). Projeto de Algoritmos - Fundamentos, Análise e Exemplos da Internet.
- ZIVIANI, N. (2007). Projeto e Algoritmos com implementações em Java e C++. São Paulo. Editora Thomson;

