Chapter 7

Partial Orders

Partial orders are a kind of binary relation that come up a lot. The familiar $\leq$ order on numbers is a partial order, but so is the containment relation on sets and the divisibility relation on integers.

Partial orders have particular importance in computer science because they capture key concepts used, for example, in solving task scheduling problems, analyzing concurrency control, and proving program termination.

7.1 Axioms for Partial Orders

The prerequisite structure among MIT subjects provides a nice illustration of partial orders. Here is a table indicating some of the prerequisites of subjects in the the Course 6 program of Spring '07:

Direct Prerequisites	Subject
18.01	6.042
18.01	18.02
18.01	18.03
8.01	8.02
6.001	6.034
6.042	6.046
18.03, 8.02	6.002
6.001, 6.002	6.004
6.001, 6.002	6.003
6.004	6.033
6.033	6.857
6.046	6.840

Since 18.01 is a direct prerequisite for 6.042, a student must take 18.01 before 6.042. Also, 6.042 is a direct prerequisite for 6.046, so in fact, a student has to take *both* 18.01 and 6.042 before taking 6.046. So 18.01 is also really a prerequisite for

6.046, though an implicit or indirect one; we'll indicate this by writing

$$18.01 \rightarrow 6.046.$$

This prerequisite relation has a basic property known as *transitivity*: if subject a is an indirect prerequisite of subject b, and b is an indirect prerequisite of subject c, then a is also an indirect prerequisite of c.

In this table, a longest sequence of prerequisites is

$$18.01 \rightarrow 18.03 \rightarrow 6.002 \rightarrow 6.004 \rightarrow 6.033 \rightarrow 6.857$$

so a student would need at least six terms to work through this sequence of subjects. But it would take a lot longer to complete a Course 6 major if the direct prerequisites led to a situation[1] where two subjects turned out to be prerequisites of *each other*! So another crucial property of the prerequisite relation is that if $a \rightarrow b$, then it is not the case that $b \rightarrow a$. This property is called *asymmetry*.

Another basic example of a partial order is the subset relation, $\subseteq$, on sets. In fact, we'll see that every partial order can be represented by the subset relation.

Definition 7.1.1. A binary relation, R, on a set A is:

- *transitive*   iff $[a \, R \, b \text{ and } b \, R \, c]$ IMPLIES $a \, R \, c$ for every $a, b, c \in A$,

- *asymmetric*   iff $a \, R \, b$ IMPLIES NOT$(b \, R \, a)$ for all $a, b \in A$,

- a *strict partial order* iff it is transitive and asymmetric.

So the prerequisite relation, $\rightarrow$, on subjects in the MIT catalogue is a strict partial order. More familiar examples of strict partial orders are the relation, $<$, on real numbers, and the proper subset relation, $\subset$, on sets.

The subset relation, $\subseteq$, on sets and $\leq$ relation on numbers are examples of *reflexive* relations in which each element is related to itself. Reflexive partial orders are called *weak* partial orders. Since asymmetry is incompatible with reflexivity, the asymmetry property in weak partial orders is relaxed so it applies only to two different elements. This relaxation of the asymmetry is called antisymmetry:

Definition 7.1.2. A binary relation, R, on a set A, is

- *reflexive*   iff $a \, R \, a$ for all $a \in A$,

- *antisymmetric*   iff $a \, R \, b$ IMPLIES NOT$(b \, R \, a)$ for all $a \neq b \in A$,

- a *weak partial order* iff it is transitive, reflexive and antisymmetric.

Some authors define partial orders to be what we call weak partial orders, but we'll use the phrase "partial order" to mean either a weak or strict one.

For weak partial orders in general, we often write an ordering-style symbol like $\preceq$ or $\sqsubseteq$ instead of a letter symbol like R. (General relations are usually denoted

---

[1]MIT's Committee on Curricula has the responsibility of watching out for such bugs that might creep into departmental requirements.

by a letter like R instead of a cryptic squiggly symbol, so $\preceq$ is kind of like the musical performer/composer Prince, who redefined the spelling of his name to be his own squiggly symbol. A few years ago he gave up and went back to the spelling "Prince.") Likewise, we generally use $\prec$ or $\sqsubset$ to indicate a strict partial order.

Two more examples of partial orders are worth mentioning:

Example 7.1.3. Let A be some family of sets and define $a\ R\ b$ iff $a \supset b$. Then R is a strict partial order.

For integers, m, n we write $m \mid n$ to mean that m *divides* n, namely, there is an integer, k, such that $n = km$.

Example 7.1.4. The divides relation is a weak partial order on the nonnegative integers.

7.2 Representing Partial Orders by Set Containment

Axioms can be a great way to abstract and reason about important properties of objects, but it helps to have a clear picture of the things that satisfy the axioms. We'll show that every partial order can be pictured as a collection of sets related by containment. That is, every partial order has the "same shape" as such a collection. The technical word for "same shape" is "isomorphic."

Definition 7.2.1. A binary relation, R, on a set, A, is *isomorphic* to a relation, S, on a set D iff there is a relation-preserving bijection from A to D. That is, there is bijection $f : A \to D$, such that for all $a, a' \in A$,

$$a\ R\ a' \quad \text{iff} \quad f(a)\ S\ f(a').$$

Theorem 7.2.2. *Every weak partial order, $\preceq$, is isomorphic to the subset relation, on a collection of sets.*

To picture a partial order, $\preceq$, on a set, A, as a collection of sets, we simply represent each element A by the set of elements that are $\preceq$ to that element, that is,

$$a \longleftrightarrow \{b \in A \mid b \preceq a\}.$$

For example, if $\preceq$ is the divisibility relation on the set of integers, $\{1, 3, 4, 6, 8, 12\}$, then we represent each of these integers by the set of integers in A that divides it. So

$$
\begin{aligned}
1 &\longleftrightarrow \{1\} \\
3 &\longleftrightarrow \{1, 3\} \\
4 &\longleftrightarrow \{1, 4\} \\
6 &\longleftrightarrow \{1, 3, 6\} \\
8 &\longleftrightarrow \{1, 4, 8\} \\
12 &\longleftrightarrow \{1, 3, 4, 6, 12\}
\end{aligned}
$$

So, the fact that $3 \mid 12$ corresponds to the fact that $\{1, 3\} \subseteq \{1, 3, 4, 6, 12\}$.

In this way we have completely captured the weak partial order $\preceq$ by the subset relation on the corresponding sets. Formally, we have

Lemma 7.2.3. *Let $\preceq$ be a weak partial order on a set, A. Then $\preceq$ is isomorphic to the subset relation on the collection of inverse images of elements $a \in A$ under the $\preceq$ relation.*

We leave the proof to Problem 7.3. Essentially the same construction shows that strict partial orders can be represented by set under the proper subset relation, $\subset$.

7.2.1   Problems

Class Problems

Problem 7.1.

Direct Prerequisites	Subject
18.01	6.042
18.01	18.02
18.01	18.03
8.01	8.02
8.01	6.01
6.042	6.046
18.02, 18.03, 8.02, 6.01	6.02
6.01, 6.042	6.006
6.01	6.034
6.02	6.004

(a) For the above table of MIT subject prerequisites, draw a diagram showing the subject numbers with a line going down to every subject from each of its (direct) prerequisites.

(b) Give an example of a collection of sets partially ordered by the proper subset relation, $\subset$, that is isomorphic to ("same shape as") the prerequisite relation among MIT subjects from part (a).

(c) Explain why the empty relation is a strict partial order and describe a collection of sets partially ordered by the proper subset relation that is isomorphic to the empty relation on five elements —that is, the relation under which none of the five elements is related to anything.

(d) Describe a *simple* collection of sets partially ordered by the proper subset relation that is isomorphic to the "properly contains" relation, $\supset$, on $\mathcal{P}\{1, 2, 3, 4\}$.

Problem 7.2.
Consider the proper subset partial order, $\subset$, on the power set $\mathcal{P}\{1, 2, \ldots, 6\}$.

(a) What is the size of a maximal chain in this partial order? Describe one.

(b) Describe the largest antichain you can find in this partial order.

(c) What are the maximal and minimal elements? Are they maximum and mini-mum?

(d) Answer the previous part for the $\subset$ partial order on the set $\mathcal{P}\{1, 2, \ldots, 6\} - \emptyset$.

Homework Problems

Problem 7.3.
This problem asks for a proof of Lemma 7.2.3 showing that every weak partial order can be represented by (is isomorphic to) a collection of sets partially ordered under set inclusion ($\subseteq$). Namely,

Lemma. *Let $\preceq$ be a weak partial order on a set, A. For any element $a \in A$, let*

$$L(a) ::= \{b \in A \mid b \preceq a\},$$
$$\mathcal{L} ::= \{L(a) \mid a \in A\}.$$

Then the function $L : A \to \mathcal{L}$ is an isomorphism from the $\preceq$ relation on A, to the subset relation on $\mathcal{L}$.

(a) Prove that the function $L : A \to \mathcal{L}$ is a bijection.

(b) Complete the proof by showing that

$$a \preceq b \quad \text{iff} \quad L(a) \subseteq L(b) \tag{7.1}$$

for all $a, b \in A$.

7.3 Total Orders

The familiar order relations on numbers have an important additional property: given two different numbers, one will be bigger than the other. Partial orders with this property are said to be *total*[2] *orders*.

Definition 7.3.1. Let R be a binary relation on a set, A, and let a, b be elements of A. Then a and b are *comparable* with respect to R iff [a R b OR b R a]. A partial order for which every two different elements are comparable is called a *total order*.

So $<$ and $\leq$ are total orders on $\mathbb{R}$. On the other hand, the subset relation is *not* total, since, for example, any two different finite sets of the same size will be incomparable under $\subseteq$. The prerequisite relation on Course 6 required subjects is also not total because, for example, neither 8.01 nor 6.001 is a prerequisite of the other.

---

[2]"Total" is an overloaded term when talking about partial orders: being a total order is a much stronger condition than being a partial order that is a total relation. For example, any weak partial order such as $\subseteq$ is a total relation.

7.3.1   Problems

Practice Problems

Problem 7.4.
For each of the binary relations below, state whether it is a strict partial order, a weak partial order, or neither. If it is not a partial order, indicate which of the axioms for partial order it violates. If it is a partial order, state whether it is a total order and identify its maximal and minimal elements, if any.

(a) The superset relation, $\supseteq$ on the power set $\mathcal{P}\{1, 2, 3, 4, 5\}$.

(b) The relation between any two nonnegative integers, a, b that the remainder of a divided by 8 equals the remainder of b divided by 8.

(c) The relation between propositional formulas, G, H, that G IMPLIES H is valid.

(d) The relation 'beats' on Rock, Paper and Scissor (for those who don't know the game Rock, Paper, Scissors, Rock beats Scissors, Scissors beats Paper and Paper beats Rock).

(e) The empty relation on the set of real numbers.

(f) The identity relation on the set of integers.

(g) The divisibility relation on the integers, $\mathbb{Z}$.

Class Problems

Problem 7.5. (a) Verify that the divisibility relation on the set of nonnegative integers is a weak partial order.

(b) What about the divisibility relation on the set of integers?

Problem 7.6.
Consider the nonnegative numbers partially ordered by divisibility.

(a) Show that this partial order has a unique minimal element.

(b) Show that this partial order has a unique maximal element.

(c) Prove that this partial order has an infinite chain.

(d) An *antichain* in a partial order is a set of elements such that any two elements in the set are incomparable. Prove that this partial order has an infinite antichain. *Hint:* The primes.

(e) What are the minimal elements of divisibility on the integers greater than 1? What are the maximal elements?

Problem 7.7.
How many binary relations are there on the set $\{0, 1\}$?

How many are there that are transitive?, ... asymmetric?, ... reflexive?, ... irreflexive?, ... strict partial orders?, ... weak partial orders?

Hint: There are easier ways to find these numbers than listing all the relations and checking which properties each one has.

Problem 7.8.
A binary relation, R, on a set, A, is *irreflexive* iff NOT$(a\ R\ a)$ for all $a \in A$. Prove that if a binary relation on a set is transitive and irreflexive, then it is strict partial order.

Problem 7.9.
Prove that if R is a partial order, then so is R^{-1}

Homework Problems

Problem 7.10.
Let R and S be binary relations on the same set, A.

Definition 7.3.2. The *composition*, $S \circ R$, of R and S is the binary relation on A defined by the rule:[3]

$$a\ (S \circ R)\ c \quad \text{iff} \quad \exists b\,[a\ R\ b\ \text{AND}\ b\ S\ c].$$

Suppose both R and S are transitive. Which of the following new relations must also be transitive? For each part, justify your answer with a brief argument if the new relation is transitive and a counterexample if it is not.

(a) R^{-1}

(b) $R \cap S$

(c) $R \circ R$

(d) $R \circ S$

Exam Problems

Problem 7.11.

---

[3]Note the reversal in the order of R and S. This is so that relational composition generalizes function composition, Composing the functions f and g means that f is applied first, and then g is applied to the result. That is, the value of the composition of f and g applied to an argument, x, is $g(f(x))$. To reflect this, the notation $g \circ f$ is commonly used for the composition of f and g. Some texts do define $g \circ f$ the other way around.

(a) For each row in the following table, indicate whether the binary relation, R, on the set, A, is a weak partial order or a total order by filling in the appropriate entries with either Y = YES or N = NO. In addition, list the minimal and maximal elements for each relation.

A	a R b	weak partial order	total order	minimal(s)	maximal(s)
$\mathbb{R} - \mathbb{R}^+$	$a \mid b$				
$\mathcal{P}(\{1,2,3\})$	$a \subseteq b$				
$\mathbb{N} \cup \{i\}$	$a > b$				

(b) What is the longest *chain* on the subset relation, $\subseteq$, on $P(\{1,2,3\})$? (If there is more than one, provide ONE of them.)

(c) What is the longest *antichain* on the subset relation, $\subseteq$, on $P(\{1,2,3\})$? (If there is more than one, provide *one* of them.)

7.4   Product Orders

Taking the product of two relations is a useful way to construct new relations from old ones.

Definition 7.4.1. The product, $R_1 \times R_2$, of relations R_1 and R_2 is defined to be the relation with

$$\text{domain}\,(R_1 \times R_2) \quad ::= \quad \text{domain}\,(R_1) \times \text{domain}\,(R_2),$$
$$\text{codomain}\,(R_1 \times R_2) \quad ::= \quad \text{codomain}\,(R_1) \times \text{codomain}\,(R_2),$$
$$(a_1, a_2)\,(R_1 \times R_2)\,(b_1, b_2) \quad \text{iff} \quad [a_1\,R_1\,b_1 \text{ and } a_2\,R_2\,b_2].$$

Example 7.4.2. Define a relation, Y, on age-height pairs of being younger *and* shorter. This is the relation on the set of pairs (y, h) where y is a nonnegative integer ≤ 2400

which we interpret as an age in months, and h is a nonnegative integer ≤ 120 describing height in inches. We define Y by the rule

$$(y_1, h_1)\, Y\, (y_2, h_2) \quad \text{iff} \quad y_1 \leq y_2 \text{ AND } h_1 \leq h_2.$$

That is, Y is the product of the $\leq$-relation on ages and the $\leq$-relation on heights.

It follows directly from the definitions that products preserve the properties of transitivity, reflexivity, irreflexivity, and antisymmetry, as shown in Problem 7.12. That is, if R_1 and R_2 both have one of these properties, then so does $R_1 \times R_2$. This implies that if R_1 and R_2 are both partial orders, then so is $R_1 \times R_2$.

On the other hand, the property of being a total order is not preserved. For example, the age-height relation Y is the product of two total orders, but it is not total: the age 240 months, height 68 inches pair, (240,68), and the pair (228,72) are incomparable under Y.

7.4.1 Problems

Class Problems

Problem 7.12.
Let R_1, R_2 be binary relations on the same set, A. A relational property is preserved under product, if $R_1 \times R_2$ has the property whenever both R_1 and R_2 have the property.

 (a) Verify that each of the following properties are preserved under product.

   1. reflexivity,

   2. antisymmetry,

   3. transitivity.

 (b) Verify that if *either* of R_1 or R_2 is irreflexive, then so is $R_1 \times R_2$.

Note that it now follows immediately that if if R_1 and R_2 are partial orders and at least one of them is strict, then $R_1 \times R_2$ is a strict partial order.

7.5 Scheduling

Scheduling problems are a common source of partial orders: there is a set, A, of tasks and a set of constraints specifying that starting a certain task depends on other tasks being completed beforehand. We can picture the constraints by drawing labelled boxes corresponding to different tasks, with an arrow from one box to another if the first box corresponds to a task that must be completed before starting the second one.

Example 7.5.1. Here is a drawing describing the order in which you could put on clothes. The tasks are the clothes to be put on, and the arrows indicate what should be put on directly before what.
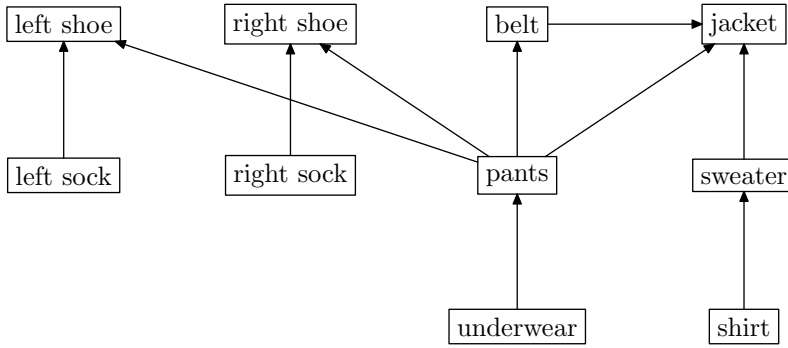
When we have a partial order of tasks to be performed, it can be useful to have an order in which to perform all the tasks, one at a time, while respecting the dependency constraints. This amounts to finding a total order that is consistent with the partial order. This task of finding a total ordering that is consistent with a partial order is known as *topological sorting*.

Definition 7.5.2. A *topological sort* of a partial order, $\prec$, on a set, A, is a total ordering, $\sqsubset$, on A such that

$$a \prec b \text{ IMPLIES } a \sqsubset b.$$

For example,

$$\text{shirt} \sqsubset \text{sweater} \sqsubset \text{underwear} \sqsubset \text{leftsock} \sqsubset \text{rightsock} \sqsubset \text{pants}$$
$$\sqsubset \text{leftshoe} \sqsubset \text{rightshoe} \sqsubset \text{belt} \sqsubset \text{jacket},$$

is one topological sort of the partial order of dressing tasks given by Example 7.5.1; there are several other possible sorts as well.

Topological sorts for partial orders on finite sets are easy to construct by starting from *minimal* elements:

Definition 7.5.3. Let $\preceq$ be a partial order on a set, A. An element $a_0 \in A$ is *minimum* iff it is $\preceq$ every other element of A, that is, $a_0 \preceq b$ for all $b \neq a_0$.

The element a_0 is *minimal* iff no other element is $\preceq a_0$, that is, $\text{NOT}(b \preceq a_0)$ for all $b \neq a_0$.

There are corresponding definitions for *maximum* and *maximal*. Alternatively, a maximum(al) element for a relation, R, could be defined to be as a minimum(al) element for R^{-1}.

In a total order, minimum and minimal elements are the same thing. But a partial order may have no minimum element but lots of minimal elements. There are four minimal elements in the clothes example: leftsock, rightsock, underwear, and shirt.

To construct a total ordering for getting dressed, we pick one of these minimal elements, say shirt. Next we pick a minimal element among the remaining ones. For example, once we have removed shirt, sweater becomes minimal. We continue in this way removing successive minimal elements until all elements have been picked. The sequence of elements in the order they were picked will be a topological sort. This is how the topological sort above for getting dressed was constructed.

So our construction shows:

Theorem 7.5.4. *Every partial order on a finite set has a topological sort.*

There are many other ways of constructing topological sorts. For example, instead of starting "from the bottom" with minimal elements, we could build a total starting *anywhere* and simply keep putting additional elements into the total order wherever they will fit. In fact, the domain of the partial order need not even be finite: we won't prove it, but *all* partial orders, even infinite ones, have topological sorts.

7.5.1 Parallel Task Scheduling

For a partial order of task dependencies, topological sorting provides a way to execute tasks one after another while respecting the dependencies. But what if we have the ability to execute more than one task at the same time? For example, say tasks are programs, the partial order indicates data dependence, and we have a parallel machine with lots of processors instead of a sequential machine with only one. How should we schedule the tasks? Our goal should be to minimize the total *time* to complete all the tasks. For simplicity, let's say all the tasks take the same amount of time and all the processors are identical.

So, given a finite partially ordered set of tasks, how long does it take to do them all, in an optimal parallel schedule? We can also use partial order concepts to analyze this problem.

In the clothes example, we could do all the minimal elements first (leftsock, rightsock, underwear, shirt), remove them and repeat. We'd need lots of hands, or maybe dressing servants. We can do pants and sweater next, and then leftshoe, rightshoe, and belt, and finally jacket.

In general, a *schedule* for performing tasks specifies which tasks to do at successive steps. Every task, a, has be scheduled at some step, and all the tasks that have to be completed before task a must be scheduled for an earlier step.

Definition 7.5.5. A *parallel schedule* for a strict partial order, $\prec$, on a set, A, is a

partition[4] of A into sets $A_0, A_1, \ldots$, such that for all $a, b \in A$, $k \in \mathbb{N}$,

$$[a \in A_k \text{ AND } b \prec a] \quad \text{IMPLIES} \quad b \in A_j \text{ for some } j < k.$$

The set A_k is called the set of elements *scheduled at step* k, and the *length* of the schedule is the number of sets A_k in the partition. The maximum number of elements scheduled at any step is called the *number of processors* required by the schedule.

So the schedule we chose above for clothes has four steps

$$A_0 = \{\text{leftsock, rightsock, underwear, shirt}\},$$
$$A_1 = \{\text{pants, sweater}\},$$
$$A_2 = \{\text{leftshoe, rightshoe, belt}\},$$
$$A_3 = \{\text{jacket}\}.$$

and requires four processors (to complete the first step).

Notice that the dependencies constrain the tasks underwear, pants, belt, and jacket to be done in sequence. This implies that at least four steps are needed in *every* schedule for getting dressed, since if we used fewer than four steps, two of these tasks would have to be scheduled at the same time. A set of tasks that must be done in sequence like this is called a *chain*.

Definition 7.5.6. A *chain* in a partial order is a set of elements such that any two different elements in the set are comparable. A chain is said to *end at* an its maximum element.

In general, the earliest step at which an element a can ever be scheduled must be at least as large as any chain that ends at a. A *largest* chain ending at a is called a *critical path* to a, and the size of the critical path is called the *depth* of a. So in any possible parallel schedule, it takes at least depth (a) steps to complete task a.

There is a very simple schedule that completes every task in this minimum number of steps. Just use a "greedy" strategy of performing tasks as soon as possible. Namely, schedule all the elements of depth k at step k. That's how we found the schedule for getting dressed given above.

Theorem 7.5.7. *Let $\prec$ be a strict partial order on a set, A. A minimum length schedule for $\prec$ consists of the sets $A_0, A_1, \ldots$, where*

$$A_k ::= \{a \mid \text{depth}\,(a) = k\}.$$

---

[4]Partitioning a set, A, means "cutting it up" into non-overlapping, nonempty pieces. The pieces are called the blocks of the partition. More precisely, a *partition* of A is a set $\mathcal{B}$ whose elements are nonempty subsets of A such that

- if $B, B' \in \mathcal{B}$ are different sets, then $B \cap B' = \emptyset$, and
- $\bigcup_{B \in \mathcal{B}} B = A$.

We'll leave to Problem 7.19 the proof that the sets A_k are a parallel schedule according to Definition 7.5.5.

The minimum number of steps needed to schedule a partial order, $\prec$, is called the *parallel time* required by $\prec$, and a largest possible chain in $\prec$ is called a *critical path* for $\prec$. So we can summarize the story above by this way: with an unlimited number of processors, the minimum parallel time to complete all tasks is simply the size of a critical path:

Corollary 7.5.8. *Parallel time = length of critical path.*

7.6 Dilworth's Lemma

Definition 7.6.1. An *antichain* in a partial order is a set of elements such that any two elements in the set are incomparable.

Our conclusions about scheduling also tell us something about antichains.

Corollary 7.6.2. *If the largest chain in a partial order on a set, A, is of size t, then A can be partitioned into t antichains.*

Proof. Let the antichains be the sets $A_k ::= \{a \mid \operatorname{depth}(a) = k\}$. It is an easy exercise to verify that each A_k is an antichain (Problem 7.19) ∎

Corollary 7.6.2 implies a famous result[5] about partially ordered sets:

Lemma 7.6.3 (Dilworth)**.** *For all $t > 0$, every partially ordered set with n elements must have either a chain of size greater than t or an antichain of size at least n/t.*

Proof. Assume there is no chain of size greater than t, that is, the largest chain is of size $\leq t$. Then by Corollary 7.6.2, the n elements can be partitioned into at most t antichains. Let ℓ be the size of the largest antichain. Since every element belongs to exactly one antichain, and there are at most t antichains, there can't be more than ℓt elements, namely, $\ell t \geq n$. So there is an antichain with at least $\ell \geq n/t$ elements. ∎

Corollary 7.6.4. *Every partially ordered set with n elements has a chain of size greater than $\sqrt{n}$ or an antichain of size at least $\sqrt{n}$.*

Proof. Set $t = \sqrt{n}$ in Lemma 7.6.3. ∎

Example 7.6.5. In the dressing partially ordered set, $n = 10$.
 Try $t = 3$. There is a chain of size 4.
 Try $t = 4$. There is no chain of size 5, but there is an antichain of size $4 \geq 10/4$.

---

[5]Lemma 7.6.3 also follows from a more general result known as Dilworth's Theorem which we will not discuss.

Example 7.6.6. Suppose we have a class of 101 students. Then using the product partial order, Y, from Example 7.4.2, we can apply Dilworth's Lemma to conclude that there is a chain of 11 students who get taller as they get older, or an antichain of 11 students who get taller as they get younger, which makes for an amusing in-class demo.

7.6.1 Problems

Practice Problems

Problem 7.13.
What is the size of the longest chain that is guaranteed to exist in any partially ordered set of n elements? What about the largest antichain?

Problem 7.14.
Describe a sequence consisting of the integers from 1 to 10,000 in some order so that there is no increasing or decreasing subsequence of size 101.

Problem 7.15.
What is the smallest number of partially ordered tasks for which there can be more than one minimum time schedule? Explain.

Class Problems

Problem 7.16.
The table below lists some prerequisite information for some subjects in the MIT Computer Science program (in 2006). This defines an indirect prerequisite relation, $\prec$, that is a strict partial order on these subjects.

$$18.01 \rightarrow 6.042 \qquad\qquad 18.01 \rightarrow 18.02$$
$$18.01 \rightarrow 18.03 \qquad\qquad 6.046 \rightarrow 6.840$$
$$8.01 \rightarrow 8.02 \qquad\qquad 6.001 \rightarrow 6.034$$
$$6.042 \rightarrow 6.046 \qquad\qquad 18.03, 8.02 \rightarrow 6.002$$
$$6.001, 6.002 \rightarrow 6.003 \qquad\qquad 6.001, 6.002 \rightarrow 6.004$$
$$6.004 \rightarrow 6.033 \qquad\qquad 6.033 \rightarrow 6.857$$

(a) Explain why exactly six terms are required to finish all these subjects, if you can take as many subjects as you want per term. Using a *greedy* subject selection strategy, you should take as many subjects as possible each term. Exhibit your complete class schedule each term using a greedy strategy.

(b) In the second term of the greedy schedule, you took five subjects including 18.03. Identify a set of five subjects not including 18.03 such that it would be possi-

ble to take them in any one term (using some nongreedy schedule). Can you figure out how many such sets there are?

(c) Exhibit a schedule for taking all the courses —but only one per term.

(d) Suppose that you want to take all of the subjects, but can handle only two per term. Exactly how many terms are required to graduate? Explain why.

(e) What if you could take three subjects per term?

Problem 7.17.
A pair of 6.042 TAs, Liz and Oscar, have decided to devote some of their spare time this term to establishing dominion over the entire galaxy. Recognizing this as an ambitious project, they worked out the following table of tasks on the back of Oscar's copy of the lecture notes.

1. **Devise a logo** and cool imperial theme music - 8 days.

2. **Build a fleet** of Hyperwarp Stardestroyers out of eating paraphernalia swiped from Lobdell - 18 days.

3. **Seize control** of the United Nations - 9 days, after task #1.

4. **Get shots** for Liz's cat, Tailspin - 11 days, after task #1.

5. **Open a Starbucks chain** for the army to get their caffeine - 10 days, after task #3.

6. **Train an army** of elite interstellar warriors by dragging people to see *The Phantom Menace* dozens of times - 4 days, after tasks #3, #4, and #5.

7. **Launch the fleet** of Stardestroyers, crush all sentient alien species, and establish a Galactic Empire - 6 days, after tasks #2 and #6.

8. **Defeat Microsoft** - 8 days, after tasks #2 and #6.

We picture this information in Figure 7.1 below by drawing a point for each task, and labelling it with the name and weight of the task. An edge between two points indicates that the task for the higher point must be completed before beginning the task for the lower one.

(a) Give some valid order in which the tasks might be completed.

Liz and Oscar want to complete all these tasks in the shortest possible time. However, they have agreed on some constraining work rules.

- Only one person can be assigned to a particular task; they can not work together on a single task.
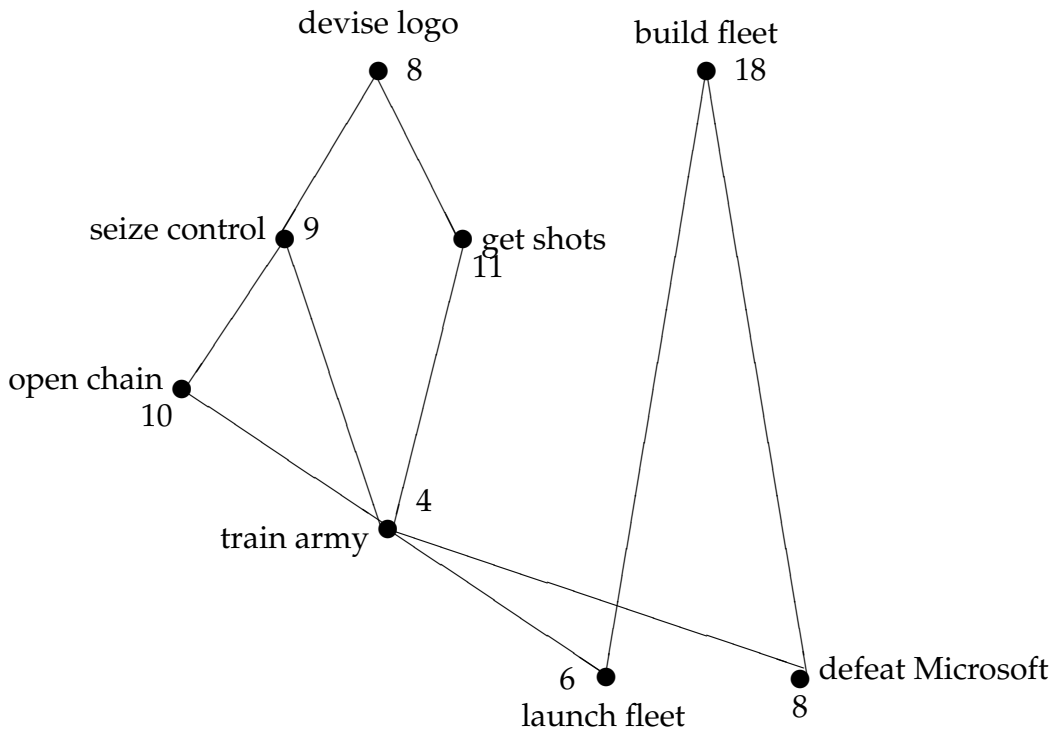
Figure 7.1: Graph representing the task precedence constraints.

- Once a person is assigned to a task, that person must work exclusively on the assignment until it is completed. So, for example, Liz cannot work on building a fleet for a few days, run to get shots for Tailspin, and then return to building the fleet.

(b) Liz and Oscar want to know how long conquering the galaxy will take. Oscar suggests dividing the total number of days of work by the number of workers, which is two. What lower bound on the time to conquer the galaxy does this give, and why might the actual time required be greater?

(c) Liz proposes a different method for determining the duration of their project. He suggests looking at the duration of the "critical path", the most time-consuming sequence of tasks such that each depends on the one before. What lower bound does this give, and why might it also be too low?

(d) What is the minimum number of days that Liz and Oscar need to conquer the galaxy? No proof is required.

Problem 7.18. (a) What are the maxim*al* and minim*al* elements, if any, of the power set $\mathcal{P}(\{1, \ldots, n\})$, where n is a positive integer, under the *empty relation*?

(b) What are the maxim*al* and minim*al* elements, if any, of the set, $\mathbb{N}$, of all non-negative integers under divisibility? Is there a minim*um* or maxim*um* element?

(c) What are the minimal and maximal elements, if any, of the set of integers greater than 1 under divisibility?

(d) Describe a partially ordered set that has no minimal or maximal elements.

(e) Describe a partially ordered set that has a *unique minimal* element, but no minimum element. *Hint:* It will have to be infinite.

Homework Problems

Problem 7.19.
Let $\prec$ be a partial order on a set, A, and let

$$A_k ::= \{a \mid \text{depth}\,(a) = k\}$$

where $k \in \mathbb{N}$.
(a) Prove that $A_0, A_1, \ldots$ is a parallel schedule for $\prec$ according to Definition 7.5.5.

(b) Prove that A_k is an antichain.

Problem 7.20.
Let S be a sequence of n different numbers. A *subsequence* of S is a sequence that can be obtained by deleting elements of S.

For example, if
$$S = (6, 4, 7, 9, 1, 2, 5, 3, 8)$$

Then 647 and 7253 are both subsequences of S (for readability, we have dropped the parentheses and commas in sequences, so 647 abbreviates $(6, 4, 7)$, for example).

An *increasing subsequence* of S is a subsequence of whose successive elements get larger. For example, 1238 is an increasing subsequence of S. Decreasing subsequences are defined similarly; 641 is a decreasing subsequence of S.

(a) List all the maximum length increasing subsequences of S, and all the maximum length decreasing subsequences.

Now let A be the *set* of numbers in S. (So $A = \{1, 2, 3, \ldots, 9\}$ for the example above.) There are two straightforward ways to totally order A. The first is to order its elements numerically, that is, to order A with the $<$ relation. The second is to order the elements by which comes first in S; call this order $<_S$. So for the example above, we would have

$$6 <_S 4 <_S 7 <_S 9 <_S 1 <_S 2 <_S 5 <_S 3 <_S 8$$

Next, define the partial order $\prec$ on A defined by the rule

$$a \prec a' \quad ::= \quad a < a' \text{ and } a <_S a'.$$

(It's not hard to prove that $\prec$ is strict partial order, but you may assume it.)

(b) Draw a diagram of the partial order, $\prec$, on A. What are the maximal elements,... the minimal elements?

(c) Explain the connection between increasing and decreasing subsequences of S, and chains and anti-chains under $\prec$.

(d) Prove that every sequence, S, of length n has an increasing subsequence of length greater than $\sqrt{n}$ or a decreasing subsequence of length at least $\sqrt{n}$.

(e) (Optional, tricky) Devise an efficient procedure for finding the longest increasing and the longest decreasing subsequence in any given sequence of integers. (There is a nice one.)

Problem 7.21.
We want to schedule n partially ordered tasks.

(a) Explain why any schedule that requires only p processors must take time at least $\lceil n/p \rceil$.

(b) Let $D_{n,t}$ be the strict partial order with n elements that consists of a chain of $t - 1$ elements, with the bottom element in the chain being a prerequisite of all the remaining elements as in the following figure:
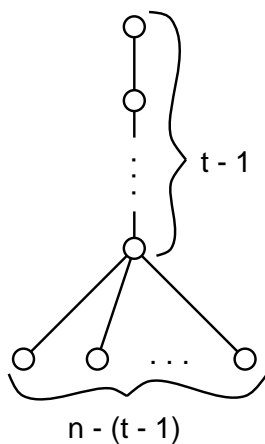
What is the minimum time schedule for $D_{n,t}$? Explain why it is unique. How many processors does it require?

(c) Write a simple formula, $M(n, t, p)$, for the minimum time of a p-processor schedule to complete $D_{n,t}$.

(d) Show that *every* partial order with n vertices and maximum chain size, t, has a p-processor schedule that runs in time $M(n, t, p)$.

Hint: Induction on t.

6.042J / 18.062J Mathematics for Computer Science
Spring 2010